BBC RD 1989/10

Research Department Report

HDTV bandwidth reduction using motion compensation and DATV

G.A. Thomas, M.A., A.M.I.E.E.

Research Department, Engineering Division
THE BRITISH BROADCASTING CORPORATION

HDTV BANDWIDTH REDUCTION USING MOTION COMPENSATION AND DATV

G. A. Thomas, M.A., A.M.I.E.E.

Summary

Recent work, described in BBC Research Department Report BBC RD 1986/5, has examined the use of a motion-adaptive bandwidth reduction system to reduce the bandwidth of a high-definition television (HDTV) signal by a factor of four. Such a reduction is required in order to enable HDTV signals to be transmitted in channels that are likely to be available in the near future.

The system was capable of transmitting a highly detailed image in stationary parts of the television picture, although the resolution of moving picture areas was reduced. A digital signal was transmitted together with the analogue picture signal to indicate to the decoder which parts of the picture were moving. Such a system has been termed Digitally Assisted Television or DATV. The loss of resolution in such areas was an objectionable artefact, which would become increasingly obvious in an HDTV transmission system as the performance of cameras and displays improved.

This Report considers the addition of motion compensation to such a bandwidth reduction system, with the aim of being able to transmit signals with high spatial resolution in all parts of the picture except those whose motion cannot be estimated accurately. Full details of the algorithm, which was developed using computer simulation, are given.

The motion-compensated bandwidth reduction system was evaluated in a series of subjective tests as a part of the Eureka 95 HDTV project. The tests examined both the quality of the decoded HDTV picture and the quality of the 'compatible' picture produced when the bandwidth-reduced signal is viewed on a MAC receiver that is not equipped with an HDTV decoder. The tests showed that the addition of motion compensation significantly improved the quality of the decoded picture at the expense of that of the compatible picture.

Issued under the Authority of

M. E. B. Moffat

Head of Research Department

Research Department, Engineering Division,
BRITISH BROADCASTING CORPORATION

(PH-294)

1989

HDTV BANDWIDTH REDUCTION USING MOTION COMPENSATION AND DATV

G. A. Thomas, M.A., A.M.I.E.E.

(PH-294)

HDTV BANDWIDTH REDUCTION USING MOTION COMPENSATION AND DATV

G. A. Thomas, M.A., A.M.I.E.E.

1. INTRODUCTION

One of the major problems of future high definition television (HDTV) systems will be that of signal distribution. An HDTV signal is likely to require at least four times the bandwidth of a conventional signal, although the channels likely to be available in the near future may not have a significantly higher capacity than a conventional channel. Thus some form of bandwidth reduction will be required.

Recent work[1] has examined the possibility of using sub-Nyquist sampling techniques to reduce the bandwidth of an HDTV signal by a factor of four. Two different types of pre-filter could be applied to the signal prior to sub-sampling; one optimum for stationary areas, the other optimum for moving ones. In stationary areas, four consecutive fields of samples were used to construct an image with high spatial resolution but poor temporal resolution. In moving areas, one field of samples was used to reconstruct each image, resulting in reduced spatial resolution but good temporal resolution. The choice of which type of filtering to use was made at the coder, by comparing the original signal with versions coded and decoded using the two alternative strategies. The method that gave the smallest coding error, accumulated over small blocks, was used to transmit the signal for that block. The two modes are referred to as the 80 ms and 20 ms branches, reflecting the period of time over which one complete image is transmitted.

The reduced bandwidth signal produced by this system consisted of an analogue part containing the sub-samples, and a digital part which told the decoder which type of reconstruction algorithm to use for each block in the picture. A television transmission system such as this, which uses both analogue picture data and digital 'assistance' data, has been termed Digitally Assisted Television, or DATV. Such a signal could, for example, be transmitted using one of the MAC (multiplexed analogue component) family of transmission formats, which has sufficient data capacity to carry the digital assistance data.

This kind of bandwidth reduction algorithm has one particularly useful property, namely that the bandwidth reduced signal, when transmitted using MAC, can be made 'compatible' with a conventional MAC signal. This means that a picture of an acceptable quality is produced when the signal is viewed on a MAC receiver without an HDTV decoder. The only visible artefact would be some moving dot patterns in detailed areas (due to the spectrum folding). A compatible HDTV transmission system such as this could be introduced in an evolutionary manner, in much the same way as colour was introduced onto monochrome television transmissions.

If compatibility were not an important factor, other coding schemes (for example, those discussed in another Report[2]) may prove more suitable. The fundamental constraint imposed by compatibility is that the data rate must be constant across the picture; non-compatible systems can proportion more of the channel capacity to active areas of the picture and hence may be able to make more efficient use of the available bandwidth.

Another useful feature of the system described in Ref. 1 is that the motion information carried by the digital assistance channel can be used to enable high quality display field rate up-conversion to be performed, without the need for expensive motion detection circuitry in each receiver. Field rate up-conversion is necessary to reduce the visibility of large-area flicker, which can be particularly noticeable on large screen displays. In order to maintain high vertical resolution without causing blurring in moving areas, it is generally necessary to use an adaptive up-conversion method requiring reliable motion information[3].

Although this bandwidth reduction system was found to work well from the point of view of artefacts, the loss of resolution in moving areas which the observer's eye could track was objectionable. As the eye tracks an object, the image of the object is rendered stationary on the retina (except for very fast motion speeds which cause the eye to make saccadic jumps). Thus the eye's requirement for spatial detail is substantially the same in such moving areas as it is in stationary areas. Although camera integration causes some loss of resolution in moving areas (which in itself is often a disadvantage), future cameras based on CCD sensors may have electronic shutters which enable them to generate sharp images in spite of motion; any bandwidth reduction system should attempt to maintain as much of this resolution as possible.

This Report describes research into the application of motion compensation to such a bandwidth

reduction system. The aim of the research was to enable the high spatial resolution (80 ms) branch to be used for all areas of the picture for which a reliable motion vector could be found. Assuming the use of a good motion estimation technique, the lower resolution branch would only be required in areas containing erratic motion and revealed or obscured background. In such areas it is likely that the human visual system can tolerate some reduction in spatial resolution, due to the eye's inability to track erratic motion and an effect known as *masking*[4]. Hence such a system should be well matched to the eye's requirements.

In contrast to the work of Ref. 1, this research was carried out by computer simulation, rather than by the construction of real-time hardware. The increased complexity of the algorithms and the larger number of unknown factors made this the only logical choice. One drawback of this approach is that it is only possible to assess the performance of the system on a small range of sequences; nevertheless over 1000 full-size frames were processed during the course of the work.

The work described here has formed part of the BBC's contribution to Eureka Project 95 (High Definition Television). The algorithm that was developed (described in Section 4) was offered as the BBC's proposal for an HD-MAC coding algorithm, and was evaluated in a series of subjective tests in May 1988. The results of these tests are discussed in Section 5.

2. THE BASIC REQUIREMENTS FOR INCORPORATING MOTION COMPENSATION

In order to be able to use the 80 ms branch for other than near-zero velocities, it is first necessary to estimate a motion vector for each part of the picture. Secondly, the filtering, sub-sampling and reconstruction processes in the high resolution path need to be modified to take account of the estimated motion.

2.1 Motion vector estimation

There are a number of requirements that a motion estimation algorithm suitable for this application should satisfy:

Firstly, it is desirable to be able to measure motion vectors to sub-pixel accuracy (i.e. to an accuracy better that one pixel per field period). Since four fields of samples are used to reconstruct each image, any significant displacement error over the period of four fields will result in impairments in the final image.

Secondly, it would be advantageous to use a technique which generates a limited number of different vectors; this will ease the task of transmitting the vector information to the decoder in the digital assistance channel, which has a limited bandwidth.

Thirdly, in the context of using motion vector information in the decoder to carry out display field rate up-conversion, it is important that the measured vectors correspond closely to the actual motion of the scene, rather than simply indicating which parts of one field look similar to parts of another.

A motion estimation technique has been investigated[5] which appears to meet these requirements and thus was chosen for this application. The technique uses a process known as phase correlation[6] to measure the dominant types of motion in the picture, and then assigns one of the measured vectors to each small block in the picture. Details of how this technique was used in this application are given in Appendix 1.

2.2 Modifications required to incorporate motion compensation

In order to understand how motion compensation can be applied to the type of bandwidth reduction system discussed above, it is necessary to understand the principle of operation of the high spatial detail (80 ms) branch. The incoming interlaced video signal was pre-filtered with spatial and temporal pre-filters* to leave the areas of the spatio—temporal spectrum shown in Fig. 1(a), and then sampled using the lattice shown in Fig. 2. In order to recover the signal, four successive fields of samples were accumulated in a frame store, to give an array of samples arranged quincunxially. The missing samples required to form an orthogonal array were generated using a spatial interpolator. The appropriate lines of this array were output, to form the required interlaced field.

For comparison, Fig. 1(b) shows the areas of the spatio—temporal spectrum that can be transmitted via the 20 ms branch, using one field of sub-samples.

One relatively simple way of incorporating motion compensation into the 80 ms branch would be to displace the accumulated array of quincunxial samples in the decoder by the motion vector of the incoming field, prior to placing the samples of the field into the array. This would mean that the picture material in the accumulated array was always

* In the hardware system constructed during the earlier work of Ref. 1, the pre-filter was in fact a vertical—temporal filter; no filtering took place that affected the horizontal domain.

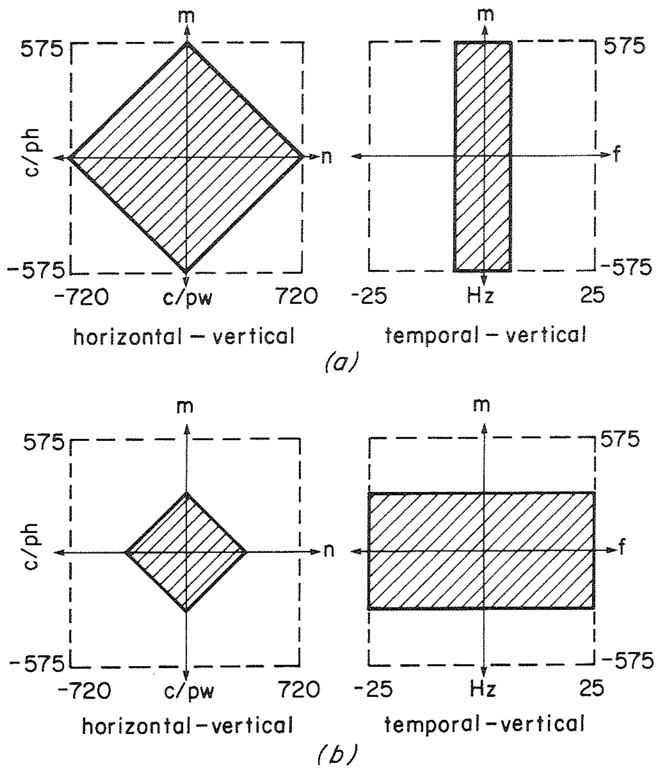

Fig. 1 - Areas of the spatio—temporal spectrum transmissible by (a) the 80 ms branch, (b) the 20 ms branch.

correctly positioned (assuming perfect motion vector measurement).

Unfortunately, this simple approach suffers from a major drawback, which may be understood by considering an example. Imagine an object moving horizontally with a velocity of one pixel per field period to the right. By the time that samples taken in the third phase of the sampling structure arrive at the decoder, the stored samples from phase one will have been displaced two pixels to the right in accordance with the motion of the object. Thus the new samples will be placed at sites already occupied by these samples. This in itself is reasonable, as the same sites on the object were sampled by both phases one and three. However, the pattern of samples arriving in the array will not accumulate to form the quincunxial pattern required for proper interpolation of the missing pixels. This would result in spatial aliasing and loss of resolution.

The fundamental cause of this problem is that the sampling structure of Fig. 2 cannot sample moving objects correctly at motion speeds other than even numbers of pixels per field period horizontally or vertically. This is exactly the problem that occurs with interlace and vertical movement in our present television system: at vertical motion speeds of an odd number of picture lines per field period, the 625/50/2:1 standard becomes effectively a 312/50/1:1 scan. The sampling structure of Fig. 2 can be considered as combining horizontal and vertical interlace.

Despite this problem it is, in theory, possible to recover the spatial resolution of the image without aliasing for all motion speeds other than those very close to an odd number of pixels per field period. This requires the use of an ideal motion-compensated temporal pre-filter prior to sampling, and a similar interpolator in the receiver, as well as perfect motion vector estimation[7].

In practice, however, such an arrangement is impractical; some residual aliasing and loss of resolution would be inevitable. The picture quality resulting from this processing would probably not be acceptable, particularly as the motion speeds that cannot be dealt with at all tend to occur frequently. One possible solution would be to change the order of the sampling sequence in a pseudo-random manner, so that the motion speeds that could not be dealt with corresponded to those of an object vibrating at random. This would have the effect of reducing the overall quality of the system for all motion speeds while preventing serious failure at certain common speeds. As such, it is not the ideal solution.

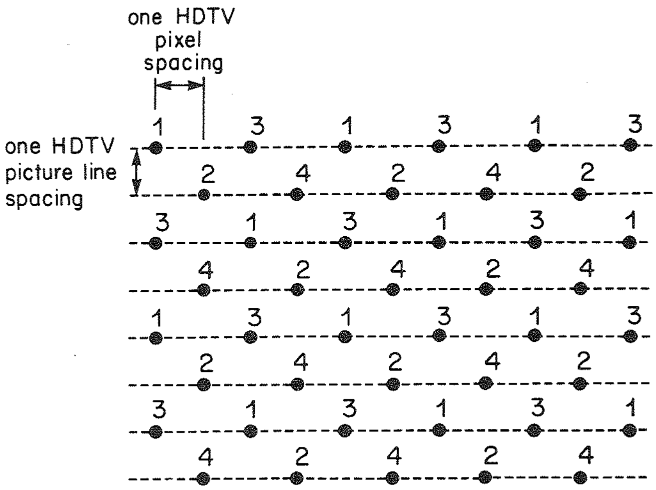

Fig. 2 - The 4-field sampling structure.

In order to maintain the same degree of spatial resolution in trackable moving areas as in stationary areas, it is thus necessary to modify the sampling lattice of Fig. 2. One way of doing this would be to use the same sampling structure but *in the reference frame of the moving object* rather than that of the camera. The remainder of the Report describes the development of this technique, and shows how it is equivalent to a low frame rate transmission system with field rate up-conversion for display. The details of the technique that was ultimately developed are explained, and the results of computer simulations are presented.

3. MOVING THE SAMPLING STRUCTURE

In order to keep the sampling structure stationary relative to the object which is being tracked, each small block in the picture must be considered to have its own sampling structure which moves according to the motion vector for that block. This section considers how such a system could be implemented in practice.

3.1 Basic principles

3.1.1 Pre-filtering

Strictly speaking, the video signal ought to be pre-filtered both spatially and temporally before sub-sampling. A spatial pre-filter is required because of the quincunxial nature of the sub-sampling structure; the pre-filter should remove diagonal frequencies as shown in Fig. 1(a). The temporal sub-sampling operation means that any temporal frequencies above 6¼ Hz that are not due to motion cannot be transmitted without aliasing, so a motion compensated temporal low pass filter, also shown in Fig. 1(a), ought to be applied.

However, both these filters were initially omitted in the simulation work. This was principally to save execution time. The omission of these filters was not thought to have serious consequences for the following reasons:

(1) As far as the spatial filter is concerned, there is hardly any energy in the diagonal frequencies outside the passband of the filter of Fig. 1(a) from scenes originated using real cameras, so the omission of this filter has little effect with such material.

(2) The vast majority of temporal variation in moving sequences is due to motion, so a motion compensated temporal filter may well have little effect. It is true that such a temporal filter would affect areas for which an incorrect motion vector had been assigned, but as mentioned previously, these areas would be transmitted using the 20 ms branch, so signals from the 80 ms branch would not be used.

The principal effect visible from omitting these filters was likely to be an increase in the noise level in picture areas sent using the 80 ms branch compared to those sent via the 20 ms branch. This suggestion was confirmed by examining the performance of the motion adaptive bandwidth reduction system built previously[1], which used a vertical−temporal filter in the 80 ms branch processing. When this filter was omitted, a slight increase in noise level was apparent. No other significant changes were visible in the picture. It is worth noting that the spatial pre-filter had never been implemented in this hardware; this had not caused significant problems.

3.1.2 Sub-sampling

Initially, the sample sites indicated in Fig. 2 for phase one were sampled. In the following field, the sampling lattice was displaced by the motion vector measured between the two fields. A spatial interpolator was used in order to be able to take account of the sub-pixel portion of the motion vector; an interlace to sequential conversion was first carried out to allow high quality inter-line interpolation to be performed. This process was repeated for successive fields, the required sampling lattice displacement being given by the sum of the motion vectors. Due to the periodic nature of the sampling structure, the displacements were summed modulo four.

In order to maintain a constant data rate, the size and shape of the blocks had to be chosen so that the number of samples within a block remained constant, regardless of the position of the sampling lattice. For example, a diamond shaped block six pixels wide (that contained 18 pixels of a picture) could contain one, two or four samples of one phase of the sampling structure, depending on the exact position of the sampling lattice. Diamond-shaped blocks eight pixels wide, however, always contain four samples; this size was chosen for the investigation.

The changes in the sampling structure had an effect on the 'compatible' signal, i.e. the sub-sampled HDTV image viewed directly on a simple receiver. In the earlier work[1], the compatible picture suffered from dot-patterning in areas of high spatial detail. With the system described here, additional impairments were introduced. The displacement of the sampling structure resulted in a small amount of 'judder' in moving parts of the scene that were being transmitted using the 80 ms branch. For example, at horizontal speeds of an odd number of pixels per field period the judder was at its worst, since sampling phases 2, 3 and 4 were moved by 1, 2 and −1 pixels respectively from their rest positions. Conversely, for horizontal speeds of an even number of pixels per field period there was no judder at all, since the sampling structure did not need to move.

3.1.3 Image reconstruction

The incoming samples were accumulated in a frame store that had the capacity to hold four fields of samples. Prior to placing a new set of samples in the store, the samples already held were displaced by the integer part of the motion vector from the previous to

the current field; this distance was the same as the integer part of the relative motion of the sampling structure in the coder. The accumulated samples were thus mis-positioned by a fraction of the sub-sample spacing; the incoming samples were also mis-positioned by the same amount because they too are forced to occupy a site on the fixed sampling lattice of the frame store. Thus the relative positions of all the accumulated samples within each block was correct, although they all had a small absolute positional error. The positional error varied from block to block, depending on the motion vectors.

The quincunxial array of samples accumulated in this way was processed using a quincunx-to-orthogonal interpolation filter, to produce an orthogonal array of samples.

The small positional error of the interpolated image was corrected for by the use of a spatial interpolator that enabled a sub-pixel offset to be added when displaying the reconstructed image. The offset varied from block to block.

One problem introduced by this fractional offset was that the quincunx-to-orthogonal interpolator operated on an array of samples that represented the image before the final sub-pixel shifts have been taken into account. This led to minor impairments at block boundaries because the aperture of the interpolator spanned regions with different offsets.

3.2 Problems with moving sampling structures

The algorithm described above was implemented on a computer-based image processing system. However, a number of problems were encountered, which required various modifications to be made. This section describes these problems, and explains why the 'moving sampling structure' approach was ultimately modified to resemble a low frame rate transmission system with field rate up-conversion for display.

3.2.1 Resetting the sampling structure

The technique of moving the sampling structure discussed in Section 3.1.2 suggests that the sampling structure moves continuously to follow motion. This is impractical for several reasons:

(a) A moving object could come to rest such that the sampling structure was displaced vertically by one picture line from its original position. This would mean that samples were always being interpolated between two field lines, resulting in reduced vertical resolution due to losses in the interlace-to-sequential conversion. (This problem would not apply if the source was sequentially scanned).

(b) The sampling structure of two adjacent blocks could move in opposite directions, resulting in a permanent gap being left in the sampling structure as a whole. This resulted in block boundaries becoming visible in some picture areas.

In order to avoid these problems, a method was devised which reset the position of the sampling structure to its 'field 1' position every fourth field. Thus the position of the structure was never more than 3 motion vectors away from where it started. However, this meant that each set of 4 fields must be dealt with in isolation, and the reconstruction process must look forwards in time as well as backwards. This implies that more field stores are required in the receiver; three field stores are sufficient for a rolling 4-field structure whereas the fixed-group structure requires seven.

This modification required that the branch selection for each block be held constant over the 4-field period, since a change of branch within a group of four fields would make it difficult to decode any of the fields in that group sent via the 80 ms branch. The method of branch selection was thus modified as follows. For every block, the coding error for both branches in each of the four fields was measured. The 80 ms branch was only selected for a particular block when it gave a lower coding error for all four fields. This avoided problems in regions of revealed or obscured background, where three of the fields were generally coded very well using the 80 ms branch, but one field (usually either the first or the last in the 4-field group) produced a large coding error.

3.2.2 Problems with revealed and obscured background

Problems were found to arise at the junction between blocks if the motion vectors of the blocks diverged. In many cases, the sampling structures moved in such a way as to leave a few sites in the frame store unfilled during image reconstruction. The solution adopted was to derive values for absent pixels from the current input sample phase, using a spatial interpolator. This effectively meant that the system could revert to a low resolution mode on a pixel by pixel basis in areas of revealed background, without the whole of the transmission block switching to the 20 ms branch.

Conversely, another problem was encountered around areas of obscured background. Samples from areas of obscured background were left in the frame

store for up to three field periods after the area disappeared from view. These samples could appear just inside the leading edge of moving objects, often causing such picture areas to revert to the 20 ms branch. This was because samples from both the object and the background were effectively competing for the same locations. This problem could be avoided by discarding samples that originated from areas not present in the current field, although this implies the use of hardware at the decoder capable of keeping a track of the picture areas in which samples originated.

3.2.3 Problems associated with motion vector accuracy

Any small inaccuracy in the motion estimation process, or effects such as an object changing its orientation or shape, were found to cause dot patterning in the processed pictures. This was the same effect as that produced by the earlier non-motion-compensated system[1] in the presence of small amounts of movement. The dot patterning, caused by mis-alignment of picture material over the 4-field period, was visible for even very small positional errors of the order of one pixel or picture line over the 4-field period.

Consideration was given to modifications that would cause the system to fail more gracefully in the presence of small vector inaccuracies. It soon became clear that the system so far developed could be implemented in a slightly different way, which not only reduced the visibility of errors, but also made the system as a whole much more elegant. These modifications are discussed in Section 4.

3.3 Summary of the moving sampling structure technique

The technique described above was simulated on an image processing system and used to process a number of short monochrome sequences, one of which formed part of a demonstration on the BBC stand at the International Broadcasting Convention in Brighton in 1986.

The application of motion compensation produced a marked improvement over the adaptive system studied earlier[1]. The improvement was particularly visible in areas of uniform motion which the observer's eye could easily track. Such areas were transmitted using the 80 ms branch for a wide range of velocities (up to the point where blurring due to camera integration made the resolution gained by the use of the motion-compensated branch negligible). However, a number of problems remained, particularly with picture material whose motion differed even slightly from that estimated. The next stage of the work attempted to improve the operation of the system by changing the approach from that based on a moving sampling structure to one based on a low frame rate transmission system.

4. A SYSTEM BASED ON 12½ Hz TRANSMISSION

4.1 Principle of operation

The object of moving the sampling structure was, as described above, to render it stationary with respect to the object being tracked.

However, it should be possible to obtain equivalent sample values by sampling at fixed sites in every fourth image (having performed an interlace to sequential conversion to obtain a complete set of vertical samples). This follows if we assume that all objects move as rigid bodies and are perfectly tracked; these assumptions are inherent in the operation of the decoder. Such a transmission system thus transmits sequentially scanned images at a 12½ Hz frame rate, and is referred to as the '12½ Hz approach' in the following discussion.

Fig. 3 shows a comparison between the coding and decoding processes using the two approaches. The only major difference in the coding processes is the sampling itself. Although the pre-filter for the 12½ Hz approach is shown to include an explicit interlace-to-sequential converter, this really does no more than the vertical–temporal filtering operation used in the other approach.

The decoding processes are very similar; the main difference being that the displacement of the image in the moving sampling structure approach is performed in two steps. Firstly, incoming samples are repositioned to place them in their correct relative position; secondly a sub-pixel interpolation is performed to account for any fractional displacement between the position of the samples in the current field and the nearest integer pixel location at which that sample is stored. In the 12½ Hz approach, both the integer and fractional parts of the shift are taken account of in one operation, the applied shift being equal to the estimated motion between the field that was sub-sampled and the output field.

This approach has a number of potential advantages over the approach outlined earlier. Firstly, any disparity between the estimated motion and the actual motion will appear as a small amount of 12½ Hz judder rather than as dot patterning, and as such may be less annoying. Indeed, it is possible to use two or more frames when carrying out the motion-compensated temporal interpolation to reduce any

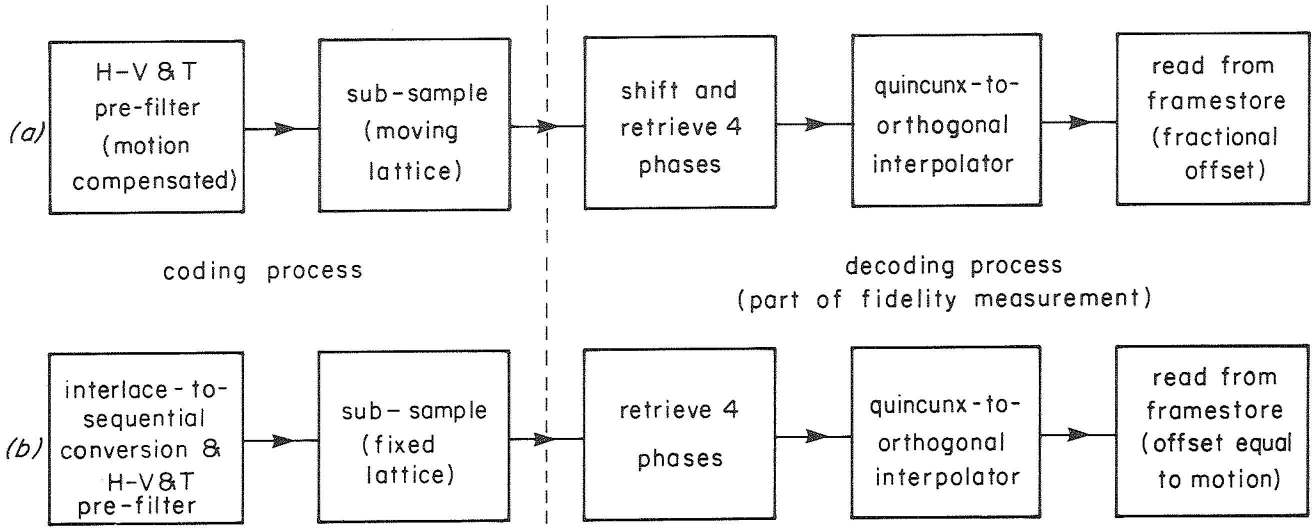

Fig. 3
Comparison of the 80 ms branch based on (a) the moving sampling structure technique, (b) 12½ Hz sequential transmission.

judder; this is discussed further in Section 4.3. Secondly, the signal does not have to pass through a motion-compensated spatial interpolator prior to sampling; this should improve picture resolution. Thirdly, it is likely to be simpler to up-convert the transmitted signal to display rates higher than 50 Hz, as sequential pictures are transmitted explicitly. One potential advantage of the use of a moving sampling structure was that the system could work on a 'rolling' 4-field aperture, giving a saving in storage requirements at the decoder. However, this could not be taken advantage of if the sampling structure was reset every fourth field as discussed earlier.

In order to maintain a degree of compatibility between the sub-sampled signal and a normal 625-line signal, the 12½ Hz samples must be repositioned prior to transmission. If this was not done, areas of the compatible picture transmitted using the 80 ms branch would be seen to judder at 12½ Hz. It is possible to reduce the level of judder to that achieved with the moving sampling structure approach, but in performing the sample repositioning, problems will arise in areas of obscured and revealed background akin to those described in Section 3.2.2.

This algorithm was indeed found to give better quality decoded pictures than those obtained with the moving sampling structure approach, and was thus fully developed into a complete proposal for a coding system. The performance of the algorithm was compared to that of a number of others in a series of subjective tests organised by Eureka 95 Project Group 5 in May 1988; the results of these tests are discussed in a later section.

The following sub-sections describe the algorithm that was developed using the 12½ Hz approach. Fig. 4 is a block diagram of the coder, omitting compensating delays but including the motion vector estimation process described in Appendix 1. Fig. 5 is a block diagram of the decoder.

The bulk of the following description applies only to the processing of the luminance component. It was not considered worthwhile to include motion compensation in the chrominance processing due to the lower chrominance resolution of the eye. The chrominance information was coded using a motion-adaptive system very similar to that used for luminance in the earlier work of Ref. 1; further details are given in Section 4.9.

4.2 Coding for the 80 ms branch

The first step in the coding process, for interlaced sources, was to perform an interlace-to-sequential conversion on the incoming signal. A motion-compensated vertical—temporal filter was used; the filter coefficients are listed in Table 1. The design of this filter is discussed in Ref. 8. The filter had six vertical taps in the central field, and five vertical taps in the fields either side. Each of these latter taps consisted of two consecutive horizontal sampling points, providing sub-pixel interpolation for horizontal movements.

Strictly speaking, a motion-compensated temporal low-pass filter should have been applied to the signal prior to sub-sampling. In practice, such a filter was found to have little effect for the reasons discussed in Section 3.1.1, and was omitted from the algorithm. However, the diagonal spatial pre-filter, also omitted from earlier work, was implemented. The filter had an aperture of 9 pixels by 9 picture lines; the coefficients are listed in Table 2. This filter was applied

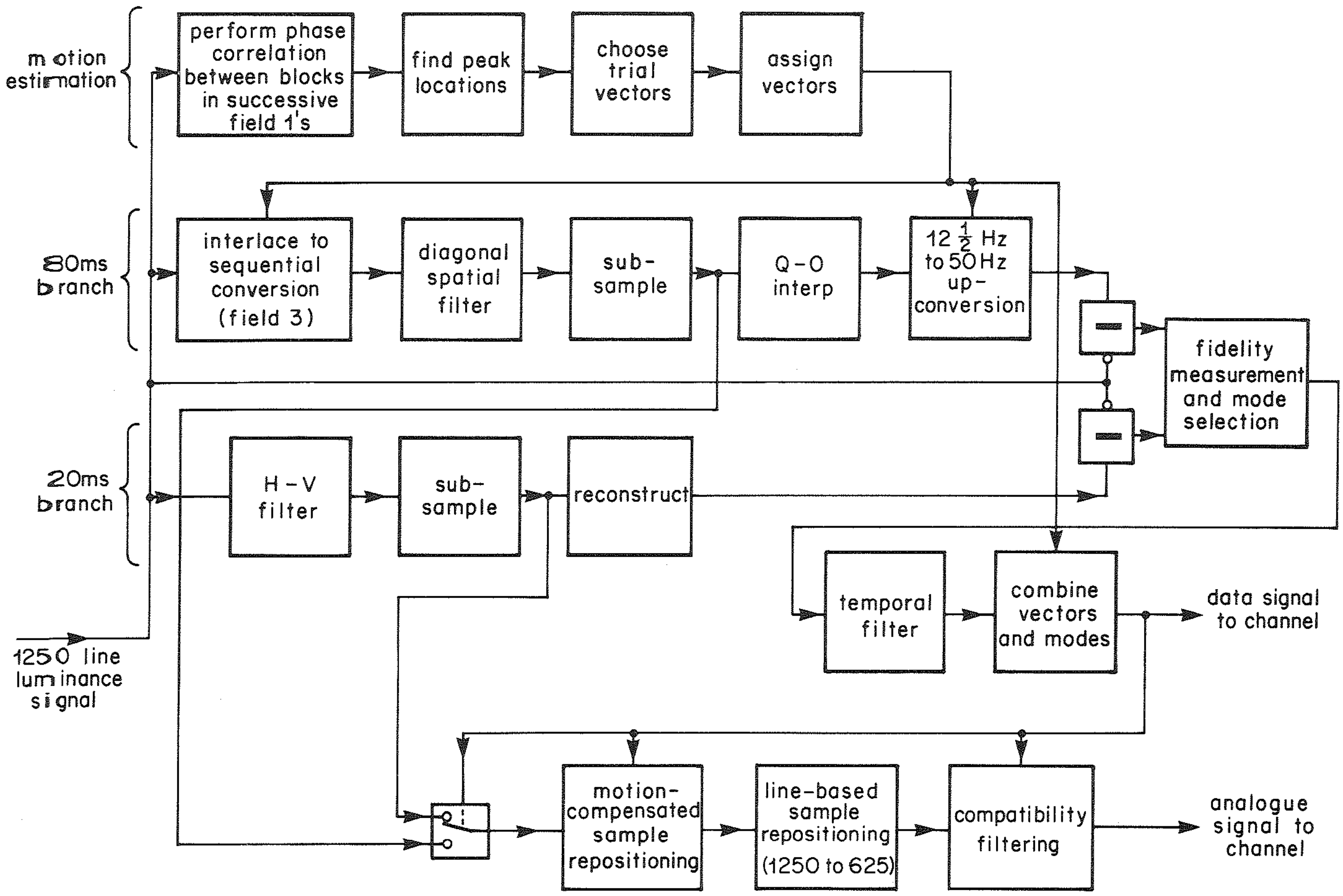

Fig. 4 - Block diagram of the bandwidth reduction encoder incorporating an 80 ms motion-compensated branch based on a 12½ Hz frame rate.

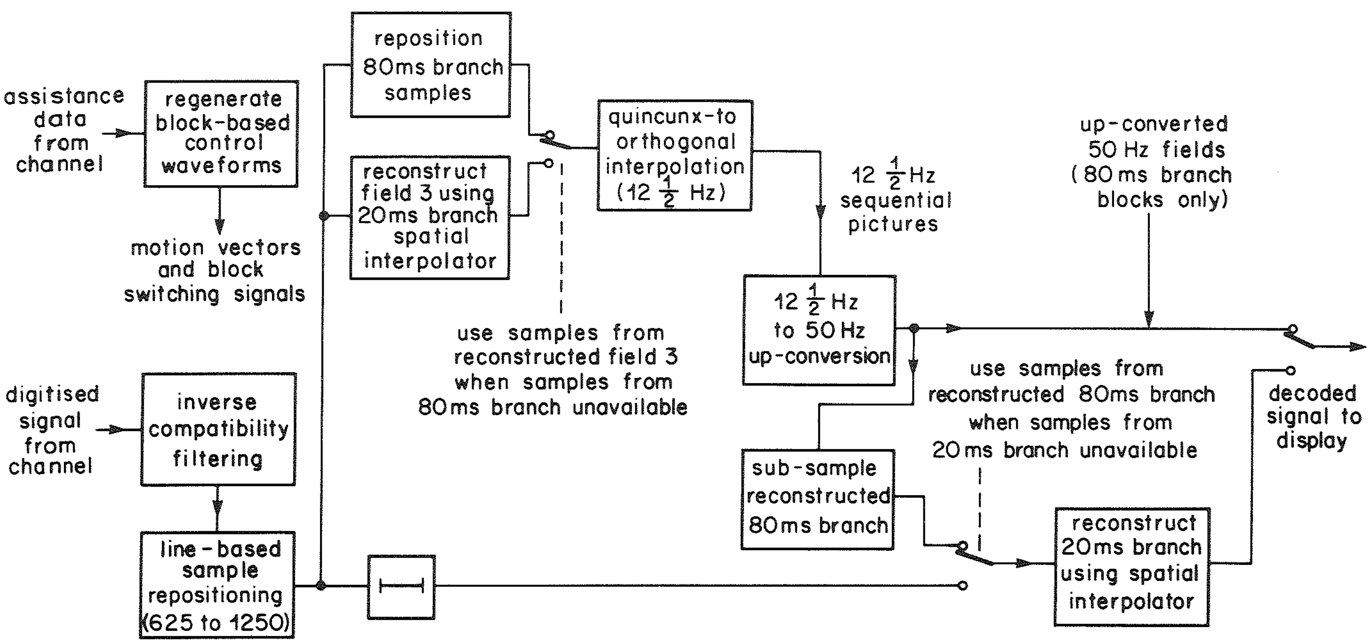

Fig. 5 - Block diagram of the bandwidth reduction decoder corresponding to the coder of Fig. 4.

Table 1
Motion-compensated interlace-to-sequential conversion filter

Coefficient arrangement:

	$a_{0,5}$	
$a_{-1,4}$		$a_{1,4}$
	$a_{0,3}$	
$a_{-1,2}$		$a_{1,2}$
	$a_{0,1}$	
$a_{-1,0}$	•	$a_{1,0}$
	$a_{0,-1}$	
$a_{-1,-2}$		$a_{1,-2}$
	$a_{0,-3}$	
$a_{-1,-4}$		$a_{1,-4}$
	$a_{0,-5}$	
preceding field	current field	following field

$a_{t,y}$ - coefficient at given vertical/temporal location
• - location of pixel being interpolated

Coefficient values as a function of motion speed:

	Motion speed (picture lines/field period):				
	0	0.25	0.5	0.75	1.0
t, y:					
−1,−4	0.0118	0.0214	0.0347	0.0332	−0.0002
−1,−2	−0.1205	−0.1331	−0.1344	−0.0867	0.0011
−1, 0	0.2917	0.2831	0.2376	0.1328	−0.0044
−1, 2	−0.1205	−0.1135	−0.1109	−0.0892	−0.0044
−1, 4	0.0118	0.0066	0.0133	0.0255	0.0010
0,−5	−0.0149	−0.0172	−0.0224	−0.0115	0.0210
0,−3	0.0374	0.0380	0.0184	−0.0466	−0.1071
0,−1	0.4165	0.4248	0.4661	0.5392	0.5851
0, 1	0.4165	0.4248	0.4661	0.5392	0.5851
0, 3	0.0374	0.0380	0.0184	−0.0466	−0.1071
0, 5	−0.0149	−0.0172	−0.0224	−0.0115	0.0210
1,−4	0.0118	0.0066	0.0133	0.0255	0.0010
1,−2	−0.1205	−0.1135	−0.1109	−0.0892	−0.0044
1, 0	0.2917	0.2831	0.2376	0.1328	−0.0044
1, 2	−0.1205	−0.1331	−0.1344	−0.0867	0.0011
1, 4	0.0118	0.0214	0.0347	0.0332	−0.0002

(Notice that the coefficients in the central field are always symmetrical about the centre. As the motion speed increases the coefficients in the adjacent fields get smaller, until at 1.0 picture lines/field period they give almost no contribution to the interpolated signal.)

For vertical motion speeds between the values listed above, linear interpolation is performed between the sets of coefficients.

Contributions from adjacent fields are obtained by horizontal interpolation from the two nearest pixels; the interpolated value is then multiplied by the appropriate coefficient.

---

to the sequential signal immediately prior to sampling; it was thus only necessary to calculate filtered values for 50% of the sites in every fourth image.

The spatially pre-filtered sequential image formed from every fourth field was sub-sampled at the quincunxially-positioned sites indicated in Fig. 2.

Table 2
Diagonal spatial filter in the 80 ms branch coder

picture lines						
	4	0	−0.0078	0	−0.0078	0
	3	0.0210	0	0.0286	0	−0.0078
	2	0	−0.0598	0	0.0286	0
	1	0.1978	0	−0.0598	0	−0.0078
	0	0.5003	0.1978	0	0.0210	0
		0	1	2	3	4
				pixels		

(These coefficients represent one out of the four symmetrical quadrants)

Table 3
Spatial pre-filter in the 20 ms branch coder

field lines								
	3	−0.0007	0.0000	0.0027	0.0062	0.0056	0.0002	−0.0016
	2	−0.0133	−0.0172	−0.0210	−0.0143	−0.0017	0.0065	0.0043
	1	0.0946	0.0695	0.0181	−0.0167	−0.0182	−0.0029	0.0027
	0	0.2227	0.1817	0.0917	0.0158	−0.0134	−0.0099	−0.0021
		0	1	2	3	4	5	6
					pixels			

(These coefficients represent one out of the four symmetrical quadrants)

Table 4
Quincunx-to-orthogonal interpolator in 80 ms branch coder

picture lines					
	4	0.006	0.007	0.012	−0.014
	3	0.001	0.003	0.018	−0.016
	2	0	−0.051	0	0.020
	1	0.197	−0.003	−0.054	0.016
	0	0.512	0.185	−0.024	0.016
		0	1	2	3
				pixels	

(These coefficients represent one out of the four symmetrical quadrants)

Table 5
Spatial interpolator in the 20 ms branch coder

field lines										
	3	0.0011	0.0003	0.0010	0.0056	0.0093	0.0060	−0.0015	−0.0055	−0.0029
	2	−0.0084	−0.0155	−0.0251	−0.0200	−0.0032	0.0084	0.0079	0.0020	−0.0007
	1	0.1007	0.0732	0.0164	−0.0221	−0.0233	−0.0056	0.0060	0.0043	0.0005
	0	0.2267	0.1843	0.0912	0.0133	−0.0155	−0.0097	−0.0006	−0.0001	−0.0005
		0	1	2	3	4	5	6	7	8
						pixels				

(These coefficients represent one out of the four symmetrical quadrants. To function as an interpolator, the filter is applied to an array in which the samples to be interpolated are set to zero. The resulting filtered array is multiplied by four to give unity gain, since three-quarters of the samples are initially set to zero.)

The motion estimation technique described in Appendix 1 was used to produce *two* sets of motion vectors for each 4-field period; each set consisting of one motion vector for each small block. The first set of vectors indicated the displacement to apply to the sampled field appropriate for the generation of the preceding two fields; the displacement for the field 40 ms prior to the sampled one being twice that for the immediately preceding field. The second set of vectors indicated the displacement to apply to the sampled field in order to generate the following two fields. Thus linear motion was assumed over a period of 40 ms, but not over 80 ms.

4.3 Coding for the 20 ms branch

The coding process for this branch consisted of a spatial pre-filter followed by a sub-sampling operation; the sites sampled were those shown in Fig. 2 that correspond to the current sampling phase. The pre-filter aperture was 13 pixels by 7 field lines; the coefficients are listed in Table 3. This filter aperture was slightly larger than that used in previous work[1].

4.4 Coding error measurement and branch selection

In order to select the transmission branch for each block, every field was reconstructed in the coder using both branches, and the relative coding errors assessed.

4.4.1 Reconstruction of the 80 ms branch

The first step in the reconstruction process was to interpolate the quincunxially-sampled frames to give an orthogonal sampling structure. This interpolation operation can be considered as a filtering operation on an array of orthogonally-positioned samples in which half of the samples (those not transmitted) have been set to zero. The filter used to perform this operation had an aperture of 7 pixels by 9 picture lines; the coefficients are listed in Table 4.

The second step involved in the 80 ms branch reconstruction was to interpolate three intermediate fields between the transmitted frames, using information from *both* of the adjacent frames.

Initial work used only the nearest frame, but subsequent work showed that a worthwhile improvement in quality could be obtained if two frames were used. This was because there was often a significant temporal component present in parts of the image after the use of motion compensation; occasionally due to small inaccuracies in motion estimates, but mainly because the assumption of rigid-body motion is not always valid. The use of ideal motion-compensated temporal pre- and post-filters would allow temporal frequencies up to 6¼ Hz to be represented without any attenuation or aliasing.

The omission of the pre-filter and the use of a simple two-tap post-filter was found to give a good compromise in terms of picture quality and complexity. The additional hardware requirements of a decoder using a 2-frame interpolator compared to one using a single frame are the duplication of a part of the circuitry; no new circuit designs are required. Of course, the use of a 2-frame interpolator in the branch decision circuitry at the coder does not preclude the use of a single frame interpolator in a simpler decoder.

Owing to the fact that the motion vectors were not constrained to be exact multiples of a pixel or picture line per field period, a spatial interpolator was needed to calculate the luminance levels at required points in the sub-sampled frames. A simple 4-point interpolator was used; interpolators with more taps were investigated and found to give only small improvements to the picture quality in highly detailed areas.

In the case of a signal originating on film replayed at 25 frames per second, the motion vectors used in the reconstruction process were modified to allow the generation of appropriately timed frames. The detection of film motion on the input signal is a relatively straightforward process and was not studied in this work. Information was included in the digital assistance data to indicate to the receiver if the signal originated from 25 Hz film.

4.4.2 Reconstruction of the 20 ms branch

The 20 ms branch was reconstructed using a spatial interpolator whose coefficients are listed in Table 5. As for the quincunx to orthogonal interpolator in the 80 ms branch, this interpolation operation can be thought of as the application of a filter to an array in which the missing samples have been set to zero. When considered as such, the aperture of the spatial interpolator was 17 pixels by 7 field lines. This aperture was slightly larger than that used in earlier work[1], as was that used in the corresponding pre-filter.

4.4.3 Branch selection

The coding error of each branch was calculated from the modulus of the difference between the reconstructed signal and the original signal. The resulting error signal was smoothed with a spatial filter having a rectangular aperture of 17 pixels by 9 field lines. This smoothed signal was then sampled at the centres of the switching blocks. The block size and shape was the same as that used in the earlier work[1], namely 12 by 12 pixel diamonds. The error filter aperture was thus slightly larger than the block size; the use of a larger aperture was found to give more

consistent decision signals. Fig. 6 shows the block shape and its relationship to the sampling lattice.

The coding errors were multiplied by weighting factors of 0.45 and 0.55 for the 80 ms and 20 ms branches respectively. This gave a small bias towards selection of the 80 ms branch.

channel according to the branch selection signal, after vertical sample repositioning to reduce the number of active lines from 1150 to 575.

However, two major impairments would be introduced in the compatible picture (in areas coded using the 80 ms branch) if the samples were used

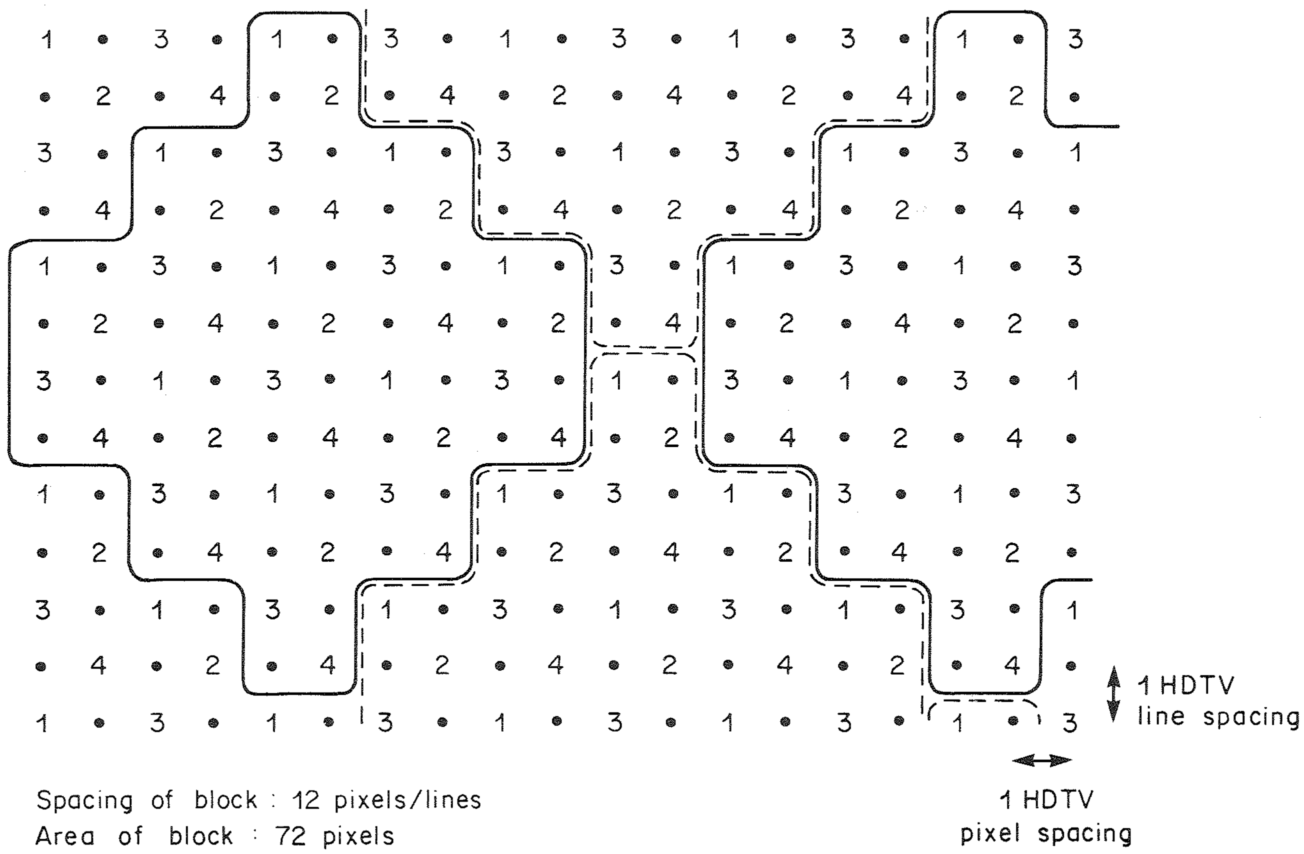

Fig. 6 - Block shape used in the transmission system and its relationship to the sampling structure.
Numbers indicate in which field the samples are transmitted.

The 80 ms branch was selected for a given block if the coding error for that branch was less than the error for the 20 ms branch in *all* of the four fields whose samples went to make up one frame.

A non-linear temporal filter was applied to this branch selection signal: any block for which the 80 ms branch was selected for only one or two successive 4-field groups was sent using the 20 ms branch. The object of this was to prevent blocks from switching into the 80 ms mode for such short periods of time that any resolution gain was outweighed by switching artifacts.

4.5 Measures to improve the compatible picture

If the quality of the compatible picture was not of importance, the samples from the two coding branches could simply be switched directly into the as they stood. The most objectionable of these would be severe 12½ Hz judder; there would also be some dot patterning due to the presence of folded spectra.

Measures were hence taken to reduce these impairments, as compatibility was considered to be an important factor for this application. Neither of these measures was entirely transparent from the point of view of the decoded picture; hence their use would not be recommended if compatibility was not a major consideration.

4.5.1 Motion-compensated sample repositioning

Prior to switching into the channel, samples in blocks chosen for transmission via the 80 ms branch were repositioned to minimise the level of 12½ Hz judder on the compatible picture.

The first stage of the repositioning process was to calculate the amount of displacement to apply. For each 80 ms block, the average motion per field in the horizontal and vertical directions was calculated by adding together the two picture-period motion vectors measured in the 4-field group, and dividing by four. The resulting vector components were rounded to the nearest even number of pixels and picture lines per field period, to obtain a displacement that could be achieved by repositioning samples in the lattice of Fig. 2.

The 12½ Hz frames were co-timed with the samples in phase 3 of the sampling structure; hence these samples never required repositioning.

Samples to be transmitted during field 1 of the 4-field sequence were taken from sites offset by twice the calculated displacement, since they were to be transmitted two field periods before the time corresponding to the field from which they originated. As both the x and y movements were always a multiple of 4, these samples always originated from sites labelled '1' in Fig. 2.

Samples to be transmitted during fields 2 and 4 were taken from sites offset by minus and plus the calculated displacement respectively. When the sum of the x and y displacements was not a multiple of 4 pixels per field period, samples transmitted during field 2 would have originated from sites labelled '4' in Fig. 2, and vice versa.

Fig. 7 shows how the samples would be repositioned in the case of horizontal motion of two pixels per field period to the right.

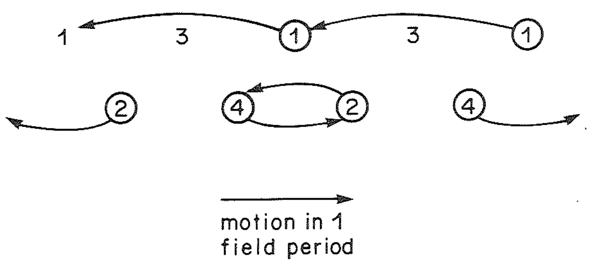

Fig. 7 - Simple example of motion-compensated sample repositioning, appropriate to a motion speed of two pixels per field period to the right.

Numbers refer to sites in sampling lattice and to the field in which the sample is transmitted.

The level of judder left after the repositioning process is shown in Fig. 8. The term *peak judder* refers to the peak positional error between where the object should be and where it is displayed. The figure also shows the level of judder in the absence of any motion-compensated sample repositioning.

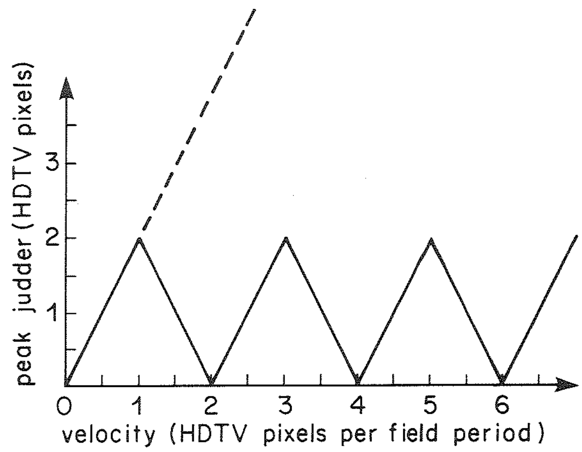

Fig. 8 - Level of 12½ Hz judder in the compatible picture with and without sample repositioning.

With sample repositioning
Without sample repositioning.

The samples that were *transmitted* at sites within a given block in the picture will generally not all have *originated* from sites within that block. At boundaries between blocks with different motion vectors or different branch selections, such samples may be discarded during the inverse repositioning process at the decoder as the samples were not required to reconstruct the block. These samples may have come, for example, from areas of revealed or obscured background that were transmitted via the 20 ms branch. A related problem was that of samples which belonged in a given block, but were not transmitted at all. In this case, the receiver is presented with insufficient samples; this problem is illustrated in Fig. 9 and will be discussed in Section 4.8.2. Both these problems are analogous to those discussed in Section 3.2.2 in connection with the moving sampling structure approach.

The motion-compensated sample repositioning process can be thought of as a crude 12½ Hz to 50 Hz up-conversion. It is performed as well as can be done without changing any sample values by interpolation,

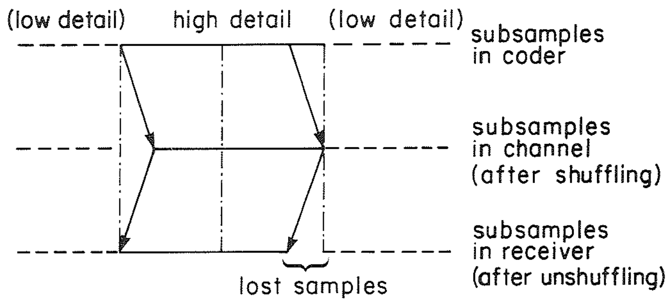

Fig. 9 - The loss of samples during the sample repositioning process.

Motion of samples during shuffling
Block boundaries.

and in such a way that it can be reversed (except at some block boundaries) and a proper up-conversion carried out in a suitable decoder.

In the case of 25 Hz film sequences, the sample repositioning was modified in order to reproduce as closely as possible the original object positioning. Thus samples transmitted in fields 1 and 2 of the 4-field sequence were displaced by the motion over a picture period (rounded to the nearest four pixels); the samples transmitted in fields 3 and 4 were not moved at all. The presence of film motion was signalled in the digital assistance signal, enabling the receiver to perform the appropriate type of inverse repositioning.

4.5.2 Filtering to reduce dot-patterning

In order to reduce the effect of dot-patterning due to folded spectra, a filter was applied to sub-samples originating from the 80 ms branch. The 4-field sampling structure of Fig. 2 gives rise to patterns that repeat at a rate of 12½ Hz; hence a suitable filter was a temporal filter with a dip in its response at this frequency. Although a spatial filter could have been used, some experiments with such filters showed that in order to achieve a sufficient reduction in the dot patterning, the subjective sharpness of the compatible picture was reduced.

The temporal filter was a 2-tap finite impulse response filter whose output was given by

$$0.75\,A(t) + 0.25\,A(t-t_p)$$

where $A(t)$ is the sub-sampled luminance signal at a time t, and t_p is one picture period. These coefficient values give a 6 dB reduction at 12½ Hz, which was found to be sufficient to reduce the visibility of the dot patterning significantly.

The filtering action was inhibited for blocks whose average motion vector over the 4-field period exceeded 1.5 pixels per field period. This was found to be necessary in order to prevent blurring on moving objects. The visibility of dot patterning in such areas was not found to be too objectionable; partially because the movement helped to mask the patterning, and also because the effect of camera integration tended to decrease the sharpness of the original image (and hence the amplitude of the folded spectra).

In theory, the effect of this filter can be reversed exactly by using a recursive filter in the decoder. In practice, noise in the transmission channel will be amplified by the the inverse filter; hence the degree of filtering must be limited in order not to incur too great a noise penalty. The figure of 6 dB attenuation was chosen by experiment as it was found to give a good compromise between compatibility improvement and noise in the decoded picture. In fact, the visibility of channel noise in blocks sent via the 20 ms branch tended to be higher than in areas sent via the 80 ms branch as the noise was of higher temporal frequency and lower spatial frequency. The addition of compatibility filtering tended to even out the noise visibility, as the filtering was applied to 80 ms branch blocks only.

4.6 Switching samples into the channel

A hard switch was performed between the repositioned and filtered 80 ms branch samples and the 20 ms branch samples according to the branch decision signal. The resulting signal was formatted as 1150 active lines of 360 samples per line. In order to present this as a signal with 575 active lines, the samples were repositioned vertically as shown in Fig. 10.

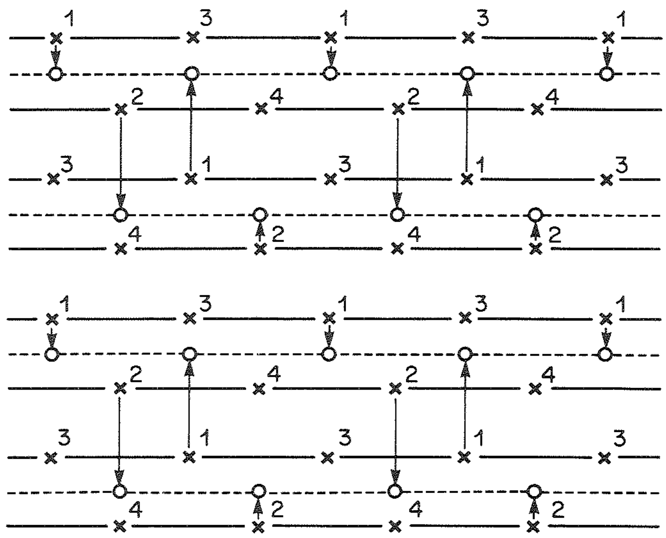

Fig. 10 - Vertical reordering of samples to produce a 625-line signal from 1250-line samples. The arrows indicate the motion of samples for fields 1 and 2 of the 4-field sequence; samples for fields 3 and 4 are moved in a similar way.

———— Source lines
-------- Output lines.

4.7 Transmission of DATV information

This sub-section discusses the digital information that had to be transmitted with the analogue signal to the decoder, and considers the requirements for a bit-rate-reduction system.

The digital assistance data consisted of a flag for every block to indicate which branch was used for transmission, and a motion vector for every block sent via the 80 ms branch. With the proposed block size (12 by 12 pixel diamonds) there were 23,000 blocks in the picture. The provisional data capacity available was about 1 Mbit/s, so the data had to be reduced to

an average of about 1.7 bits per block per frame period. Given that the raw data rate could have been as high as 14 bits per block (7 and 6 bits for vector x and y components respectively, plus one bit to indicate branch), there was clearly the need for some bit-rate reduction.

A method for reducing the bit rate of the motion vector information was suggested by the way in which the vectors were estimated. As explained in Appendix 1, each block was assigned one out of a list of eight possible 'menu' vectors, the list changing from one large 'measurement' block to the next. Thus a saving could be achieved by transmitting the vector menus separately, and sending a 3-bit number per small 'transmission' block to indicate which vector was assigned. The selection of the 20 ms branch could be represented as a ninth possible vector. This technique would reduce the data rate to just over 3 bits per block, plus some overhead for transmitting the vector menus.

The method adopted was based on this principle, with the addition of an entropy coding technique to reduce further the bit rate for the vector selection information. The vectors in each menu (including the pseudo-vector indicating selection of the 20 ms branch) were sorted into descending order of frequency of use, so that the statistical distribution of the codes for each transmission block was roughly constant across the whole picture. The transmission blocks were then assembled into groups of 16 (four horizontally, two vertically and two temporally), and the vector selection codes concatenated to make one long 'message' describing the branch selection and vectors for all of the 16 blocks. A 'codebook' was compiled of the 128,000 most frequently occurring messages; these were transmitted using short code-words; the more frequently occurring messages being allocated the shortest codewords. All remaining messages were sent in full, with a few additional bits to indicate that a full message was being sent.

This technique reduced the average bit rate for the branch and vector selection information to about 1.0 bits per block per frame (0.58 Mbit/s). More reduction could be achieved by the use of a larger codebook, although this would require the use of a larger ROM at the decoder.

The raw bit rate for each vector menu was about 100 bits per menu (eight vectors at 13 bits per vector). With the dimensions of the measurement blocks being 64 pixels by 64 picture lines, there were 432 blocks in the picture. Thus the raw rate was potentially as high as 1 Mbit/s for the menu information. However, this was reduced to an average of about 0.25 Mbit/s by transmitting information on the *difference* between successive menus, rather than transmitting each menu in full. There was generally a high degree of similarity between menus for adjacent measuring blocks because they referred to spatially adjacent areas of the picture; indeed the derivation of the eight vectors involved using vectors measured in adjacent picture areas as explained in Appendix 1.

The total data rate for the branch and vector selection information and the vector menus was thus about 0.83 Mbit/s. However, it is possible to conceive of picture material that could give rise to a rate in excess of 1 Mbit/s, for example a picture containing a large number of tracked objects moving at different speeds. There may be insufficient digital channel capacity to deal with such peak loads, so some kind of fallback system would be required. This could take the form of forcing some transmission blocks to use the 20 ms branch when the channel capacity was exceeded. If the information was transmitted for the blocks nearest the centre of the picture first, then the effect of exhausting the channel capacity before all data had been sent would be to reduce the resolution of the edges of the picture. Of course, some feedback would be required from the data coder to the sample selection circuitry, in order to override the branch decision signal.

4.8 Operation of the decoder

The operation of the decoder was very similar to that of the reconstruction process in the coder. However, a few additional steps were required. The compatibility improvement measures needed to be reversed. Also, since the apertures of the spatial and temporal interpolation filters for a given branch could extend into adjacent blocks transmitted via the other branch, interpolators were required in order to derive sample values appropriate for one branch from samples sent via the other branch.

The following sub-sections describe the operation of the decoder, concentrating on the additional processes required over those already described in Sections 4.4.1 and 4.4.2. As with the description of the coder, this explanation refers to the luminance portion of the decoder; the chrominance processing was more straightforward and is described in a later section.

A block diagram of the decoder was shown in Fig. 5.

4.8.1 Inverse compatibility filtering

The incoming samples coded using the 80 ms branch that were subject to the compatibility filtering process described in Section 4.5.2 were passed through a recursive filter to restore the signal to its original

form. The filter response was given by

$$A_{out}(t) = 1/(1-f) \; . \; A_{in}(t) - f/(1-f) \; . \; A_{out}(t-t_p)$$

where $A_{in}(t)$ is the signal from the channel,

f is a coefficient equal to 0.25 (as in the coder),

t_p is one picture period.

This filter gave a 6 dB boost at 12½ Hz.

4.8.2 Inverse sample repositioning

The next stage was to perform vertical sample repositioning in order to restore the signal with 575 active lines to a signal with twice this number. This operation was a simple inverse of the repositioning operation illustrated in Fig. 10.

The following operation was the reversal of the motion-compensated sample repositioning operation applied to blocks sent via the 80 ms branch, described in Section 4.5.1. This was achieved by reversing the entire process, so that the last sample displaced was the first to be moved back.

After the inverse repositioning process, some pixels in blocks sent via the 80 ms branch will not have received any samples. This is because the samples originating in these locations would have been moved into blocks sent via the 20 ms branch, and hence would not have been transmitted at all. The problem was illustrated in Fig. 9. In addition, there will be no samples at all in areas sent via the 20 ms branch. The motion-compensated temporal interpolation operation requires samples to be available in areas immediately adjacent to 80 ms branch blocks, as indicated in Fig. 11. It is thus necessary to generate values for all these unfilled pixels.

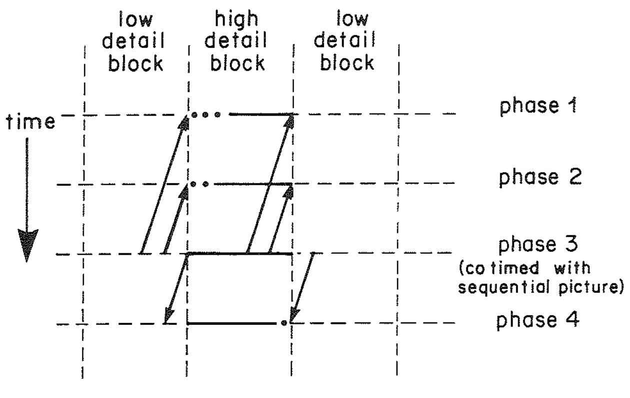

Fig. 11 - Reconstruction of the 80 ms branch, showing the need for information from adjacent blocks.

——— Areas filled with samples from 80 ms branch areas
• • • • • Areas filled with samples from 20 ms branch areas
↙ Motion vectors used in the 80 ms branch decoding process.

The missing pixels were filled with values obtained by using the 20 ms branch spatial interpolator to generate field 3 of the 4-field sequence (the field co-timed with the frame sent by the 80 ms branch). The 80 ms branch samples sent in this field are never displaced as they are transmitted at the correct time instant; thus there are never any missing samples in this field. There will inevitably be spatial aliasing in the regenerated field in areas sent via the 80 ms branch as the appropriate pre-filter was not used in these areas; however this is of little consequence as the sample values in these areas are only used to fill in small areas. This approach is the same as that described in Section 3.2.2, in connection with the 'moving sampling structure' method.

4.8.3 Reconstruction of the 80 ms branch

The sub-samples recovered from the channel, together with those generated as described above, formed a complete array of quincunxially-positioned sub-samples. This array was interpolated as described in Section 4.4.1 to give an orthogonally-sampled image.

A motion-compensated temporal interpolator was used to generate a 50 Hz interlaced signal. This used two adjacent 12½ Hz frames with a 4-point spatial interpolator in each frame, as described in Section 4.4.1. Only one frame was used for blocks that were transmitted via the 20 ms branch in one of the two adjacent frames, since signals sent via the 20 ms branch cannot be extrapolated temporally.

4.8.4 Reconstruction of the 20 ms branch

Samples sent via the 20 ms branch were not subject to any motion-compensated repositioning or filtering to improve compatibility, so no inverse processing was required.

However, the incoming samples could not be passed directly to a spatial interpolator to construct the output signal. This was because the aperture of the interpolator extended beyond the edges of the blocks sent via the 20 ms branch, into areas for which samples appropriate for this branch were not available.

The extra samples required were provided by sub-sampling the reconstructed signal from the 80 ms branch decoder. No pre-filter was applied prior to this sampling process; a degree of aliasing was not found to impair the reconstructed picture because the samples only fell in the periphery of the aperture of the interpolator.

After inclusion of these extra samples, a complete sub-sampled field was available for interpolation. The interpolator described in Section 4.4.2 was

used to generate three new samples between each pair of sub-samples, thereby generating a complete interlaced field.

The reconstructed fields from the 80 ms and 20 ms branch decoders were switched to the output according to the branch selection signal.

4.9 Chrominance processing

The chrominance information was transmitted without the use of motion compensation, using a two-branch adaptive system. The 80 ms branch corresponded to that used for luminance processing with the motion vectors set to zero; the 20 ms branch was exactly as for the luminance processing.

Prior to coding, the two chrominance signals were vertically filtered and sub-sampled in order to generate signals with half the number of active lines. This followed the format for MAC chrominance signals, in which the *U* and *V* components are transmitted on alternate field lines. A simple pre-filter was used that had an aperture of 7 field lines; the coefficients are the same as those used for down-filtering the luminance signal prior to vector measurement, and are listed in Table 6. Higher vertical resolution of the chrominance signals could have been obtained using a filter with contributions from adjacent fields.

Although this method of selecting the chrominance branch could potentially generate an inappropriate choice in picture areas containing chrominance detail without corresponding luminance detail, this was not found to be a problem in practice. Additional complexity would have been introduced by carrying out motion detection on the chrominance itself, and additional digital assistance data could potentially be required.

The chrominance sub-samples were not subjected to any filtering to improve the compatibility. The impairments visible in the chrominance components of the compatible picture were small compared to those visible in the luminance; the use of filtering would only have served to increase the level of chrominance noise in stationary areas of the decoded picture.

Decoding of the chrominance signals was straightforward, and followed the method used for the luminance signal. The absence of motion compensation led to significant simplification.

A vertical interpolator was used to generate chrominance signals for every line of the output picture from the decoded 2:1 vertically-sub-sampled signals. This interpolator was very simple, and used the same coefficients as have been suggested for use in a MAC decoder. Samples on output field lines that were coincident with sampled lines were formed with a three-tap vertical filter with coefficients of ¼, ½, ½. Samples on intermediate lines were generated with a two-tap filter with coefficients of ½, ½. As with the chrominance vertical pre-filter, better filters could have been used to improve the vertical chrominance resolution.

Table 6

Pre-filter for 2:1 vertical chrominance pre-filtering; also used for pre-filtering luminance prior to subsampling of picture for vector measurement

coefficient location: (field lines/pixels)	−3	−2	−1	0	1	2	3
coefficient value:	−1/16	0	5/16	1/2	5/16	0	−1/16

The resulting chrominance signals resembled quarter-sized luminance pictures; they were then processed using the filters and interpolators developed for the luminance component. This meant that the branch decision block size was effectively four times the area of that used for the luminance component. The use of larger blocks for chrominance was not a fundamental requirement (although it simplified the implementation of the system); smaller blocks could have been used.

The chrominance branch selection signal was derived from the motion vectors and luminance branch signal. The 80 ms branch was selected for a chrominance block if that branch was selected for all four of the corresponding luminance blocks, and the magnitude of all their motion vectors was less than 1.5 pixels per field period.

5. PERFORMANCE OF THE ALGORITHM

As has already been discussed, the motion-compensated algorithm described in Section 4 was found to give a significantly higher picture quality than the non-motion-compensated algorithm investigated previously[1]. It also gave better results than were achieved in the initial work using the technique based on moving sampling structures; more development

effort was therefore expended on the technique based on a 12½ Hz frame rate, as it appeared more promising.

The performance of the algorithm was assessed in a series of subjective tests organised as a part of the Eureka 95 HDTV project. This provided the opportunity to assess not only the performance of the algorithm itself, but also its performance relative to that of the non-motion-compensated algorithm studied previously[1], and gave a direct indication of the improvement achievable by the use of motion compensation. It was also possible to compare it to algorithms developed by other Eureka members, which had different combinations of motion-compensated and non-motion-compensated branches. Information on the background to the work within the Eureka 95 project is given in Ref. 9, and descriptions of algorithms developed by some of the other Eureka members can be found in Refs. 10-12.

5.1 Organisation of the subjective tests

The subjective tests were carried out in May 1988 by five European laboratories, including the BBC. A total of seven different algorithms were assessed using eight different test sequences. The performance of the algorithms was measured in terms of the quality of the decoded HDTV picture and that of the compatible picture.

The systems evaluated included that described in Section 4 above, and the non-motion-compensated system described in the earlier Report[1]. The other systems were developed by other European partners in the Eureka 95 project. All the systems were based on the general principles described in this Report, namely adaptive sub-sampling of the source signal with the use of a number of fields of samples to reconstruct the signal. However, the algorithm described in Section 4 was the only one that used a motion-compensated 80 ms branch, and hence the only one that was capable of maintaining the maximum spatial resolution in moving areas of the picture.

The test material consisted of eight 100-frame *YUV* sequences in the 4:2:2 625/50/2:1 format. The tests were not carried out at an HDTV standard for many reasons, including lack of universally available HDTV recorders and displays, and the length of processing time that would have been required to process such images.* The use of 625-line pictures had the advantage that the sources and displays available were mature products, capable of providing resolution up to the theoretical limits of the scanning standard; the same is not yet generally true of HDTV cameras and monitors. The processing was carried out as if the 625-line picture was a quarter-picture window taken out of a full HDTV picture.

The sequences themselves covered a wide range of picture material, originated from electronic cameras, as well as film at 25 and 50 frames/second and an HDTV camera. They are briefly described in Appendix 2.

The simulations included the addition of channel noise at a level corresponding to a carrier to noise ratio of 26 dB. This was chosen to be approximately equivalent to the noise level that might be found in a good satellite channel, taking into account the effects of non-linear pre-emphasis that was not itself simulated.

In the tests themselves, each participant was asked to grade the sequences on a continuous absolute quality scale. The participants were shown both the unprocessed and processed sequences, and the level of impairment introduced by the processing was determined by the differences between the scores. The sequences were shown in pairs, using the 'double stimulus' method.

5.2 Results of the tests

This section considers the results of the tests relating to the two BBC algorithms; the motion-compensated algorithm described in Section 4 and that described in the earlier Report[1]. The results for the algorithms developed by other organisations are not included here, as they were confidential to the other Eureka participants.

5.2.1 Decoded picture quality

Fig. 12 shows the impairments in the decoded pictures (and their standard errors) measured for the two BBC algorithms. The vertical scale represents the grade of impairment, and approximates to the CCIR 5-point impairment scale, in which a one grade impairment is termed 'perceptible but not annoying' and two grades is termed 'slightly annoying'. The horizontal axis shows the different sequences; the order of these is arbitrary.

The improvement gained by the addition of motion compensation is clear. The average impairment for the motion-compensated system was about 0.59 grades whereas that for the non-motion-compensated system was 1.34; an improvement of 0.75 grades.

* Just under two hours of computer time were required to process each 625-line picture using a MicroVAX II computer, for the algorithm described in Section 4. The total time required for all eight sequences was about nine weeks; almost nine months would have been needed if full HDTV sequences had been used.

Indeed, the average impairment for the motion-compensated system was the lowest out of all the systems that were evaluated.

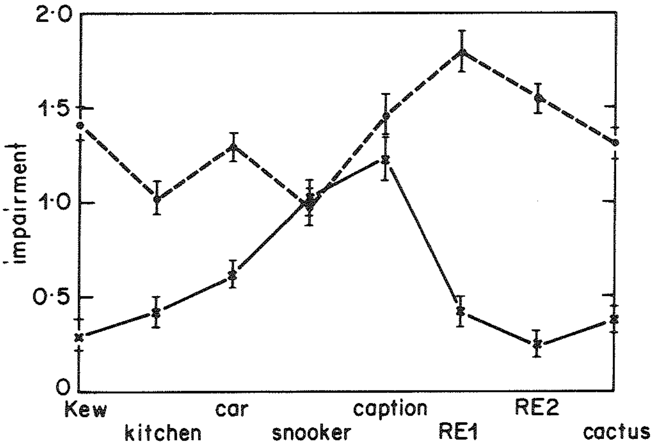

Fig. 12 - Results of the subjective tests showing the level of impairment in the decoded HDTV picture.
---- • ---- Non-motion-compensated algorithm
—— × —— Motion-compensated algorithm.

Nevertheless, the system performance varied significantly between the sequences. In particular, the results from the SNOOKER and ROLLING CAPTION sequences were noticeably worse than for the other sequences; indeed the non-motion-compensated system performed marginally better on the first of these. It is thus worth examining the operation of the algorithm on these sequences in particular, in order to pinpoint its major weaknesses.

The SNOOKER sequence (Fig. 13) showed two snooker balls rolling slowly on a track against a detailed stationary background. The balls were successfully tracked by the motion estimation algorithm and were transmitted via the 80 ms branch. However, the revealed background behind the balls was transmitted via the 20 ms branch; the loss of resolution was very noticeable due to the level of detail. The main problem was that the area transmitted with this branch changed at a rate of 12½ Hz, giving the impression that the balls were juddering. The rolling 4-field structure of the non-motion-compensated algorithm meant that no such 'jerky' motion was visible using this algorithm, although the balls always appeared blurred except for brief instants when they were stationary. Thus the main reason for the poor performance of the motion-compensated algorithm on this sequence was due to the division of the sequence into fixed groups of four fields.

The ROLLING CAPTION sequence (Fig. 14) showed electronically-generated captions moving vertically over a detailed stationary background. Some of the background was plain; in these areas the captions were successfully tracked and transmitted to good effect using the 80 ms branch. However, in many other areas the captions were transmitted using the 20 ms branch and suffered visible resolution loss. In some areas, parts of the captions were sent via the 80 ms branch, causing the background to be dragged along with the captions.

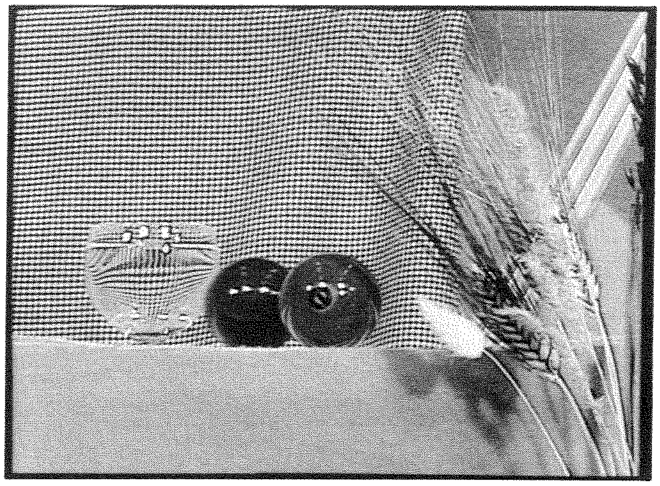

Fig. 13 - One frame from the SNOOKER test sequence.

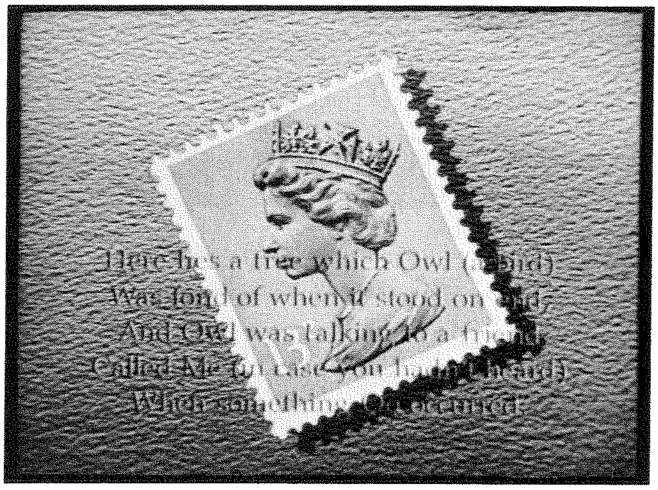

Fig. 14 - One frame from the ROLLING CAPTION test sequence.

Minor improvements could be made to the performance of the algorithm with this sequence by the use of better filters in the 20 ms branch and the chrominance processing.

However, the fundamental problem was that neither the 80 ms nor the 20 ms branch worked satisfactorily in blocks containing two highly-detailed 'objects' with significantly different motion vectors. The addition of another branch, working over a period of 40 ms, may be a solution to this problem. Such an approach has been investigated by other workers[10,11] and found to yield some advantages on this type of material. There is concern, however, that the lack of an 80 ms motion-compensated branch in

 

Fig. 15 - An example of a frame from the RENATA 1 sequence.
(a) processed by the motion-compensated coding algorithm; (b) as (a), but showing only the blocks coded using the 80 ms branch.

this other work may result in a loss of resolution in many moving areas. The best solution may therefore be to have both a 40 ms and an 80 ms motion-compensated branch.

Apart from the problems with the SNOOKER and ROLLING CAPTION sequences, the performance of the motion-compensated algorithms was generally good. An example of a decoded frame from the RENATA 1 sequence is shown in Fig. 15(a). Fig. 15(b) shows just the areas coded using the 80 ms branch; the blocks coded using the 20 ms branch are black. The use of the 20 ms branch was confined to areas containing revealed or obscured background (where motion compensation could not be used) or areas which contained little detail (where both branches performed equally well).

Incidentally, it is interesting to compare the subjective impairments measured in these tests to the RMS coding errors. These errors were obtained by calculating the square root of the sum of the squares of the differences between the grey levels of each pixel in the original and the decoded sequences, and dividing by the number of pixels in the whole sequence. This process was carried out for each of the eight test sequences for the motion-compensated algorithm. Fig. 16 shows the RMS coding errors together with the subjective impairment grades taken from Fig. 12. The RMS coding error is in units of grey levels for the quantised 8-bit signals; thus one level represents roughly 1/220 of the level of peak white. The plots have been drawn in such a way that the mean impairment grade and mean RMS coding error coincide on the vertical scales.

From Fig. 16 it is clear that the RMS coding error shows little correlation with the subjective impairment level. This was expected, and is because the two ways of measuring coding errors (subjective and objective) give different weights to different kinds of impairment. For example, a slight loss of vertical resolution can produce a significant RMS coding error, whereas the subjective effect may be negligible (it could even be beneficial when an interlaced display is

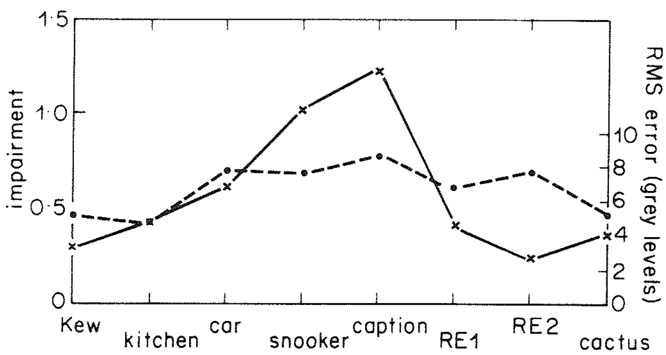

Fig. 16 - Comparison between subjective impairment and RMS coding error in the decoded picture using the motion-compensated algorithm.

—×— Impairment grade
--•-- RMS coding error.

used due to a reduction in interline twitter). This particular effect may explain the differences in the two measurements in the sequences KEW GARDENS and RENATA 2 which contain significant vertical detail.

This comparison underlines the importance of the use of subjective testing to assess system performance in this kind of application.

5.2.2 Compatible picture quality

Fig. 17 shows the impairments in the compatible pictures for the two BBC algorithms (with and without motion compensation). The standard errors are also shown. The impairment grades are relative to a compatible reference picture generated by down-filtering and sub-sampling the original sequences, and have the same significance as those for the decoded pictures.

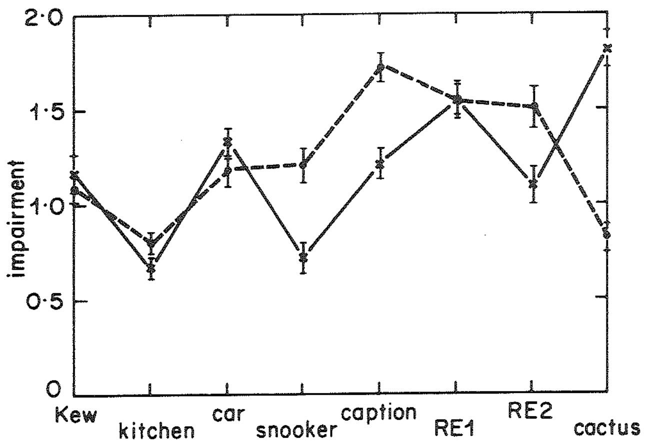

Fig. 17 - Results of the subjective tests showing the level of impairment in the compatible picture.

--•-- Non-motion-compensated algorithm (without compatibility filter)
—×— Motion-compensated algorithm (with compatibility filter)

It is unfair to make a direct comparison between the performance of the two algorithms because the non-motion-compensated algorithm did not incorporate the compatibility filter described in Section 4.5.2. This filter was omitted in order to make the system resemble that investigated earlier[1], which did not incorporate a filter.

The improvement gained by the use of the compatibility filter was sufficient to make the appearance of the compatible picture from the motion-compensated system significantly better than that of the non-motion-compensated system for a number of sequences. For example, the SNOOKER sequence contained little movement, so the impairments produced by 12½ Hz judder were small; any such impairments that were present were more than made up for by the effect of the compatibility filter.

The compatibility of the motion-compensated system was best for those scenes that contained little movement (for example SNOOKER) and the scene originating on 25 Hz film (KITCHEN); the 12½ Hz motion judder was least visible in these scenes. Conversely, the other sequences (particularly CAR, RENATA 1 and CACTUS) contained a large amount of moving detail transmitted via the 80 ms branch, resulting in significant impairments in the compatible picture. As might be expected, it was these kind of scenes that gained the most from the use of motion compensation in the decoded picture.

The motion-compensated algorithm described here gave the lowest compatible picture quality compared to the other algorithms tested. This can be attributed directly to the residual 12½ Hz judder, due to the use of a motion-compensated 80 ms branch. Algorithms using a motion-compensated 40 ms branch can be expected to show 25 Hz judder in moving areas transmitted via this branch; such judder is subjectively much less annoying due to its higher frequency and most observers are used to tolerating it in programmes originated from film.

6. SUGGESTIONS FOR FUTURE WORK

The algorithm described in this Report was found to be capable of transmitting a high quality decoded picture, producing an average impairment of about 0.6 grades for eight critical test sequences. However, this level of impairment is still higher than is desirable.

The quality of the compatible picture was disappointingly low, due almost entirely to the residual 12½ Hz judder. It was largely for this reason that the algorithm described here was not chosen for implementation in hardware by the Eureka 95 project as a proposal for an HD-MAC coding standard. The algorithm that was selected incorporated motion-compensation in a 40 ms branch rather than in an 80 ms branch, thereby producing better quality compatible pictures at the expense of a reduction in quality of the decoded picture. Such a system also requires fewer frames of storage at the decoder, and may not require motion-compensated sample repositioning, due to the higher temporal sub-sampling frequency; this avoids the problems with lost samples discussed in Section 4.8.2.

In order to retain the benefits of an 80 ms branch (no loss of spatial resolution in moving areas) without introducing unacceptable artefacts (e.g. 12½ Hz judder) in the compatible picture, it is worth looking at variations on the method described in this Report.

One possible approach would be as follows. In order to avoid the introduction of judder into the compatible picture, the sampling structure would be kept fixed, and the same number of sites would be sampled in every field. This implies that aliasing will be present in the sampled signal at certain speeds of movement; it may be possible to reduce the visibility of such aliasing by the use of motion-compensated temporal filtering at the decoder. To prevent the system failing for common speeds (such as a velocity of one pixel per field period) the phases of the sampling structure could be randomised so that the velocities for which the system fails correspond to pseudo-random motion rather than common constant velocities.

Such a system could be expected to exhibit a poorer peak performance than that described in this Report, but may produce a better compatible picture. In addition, the problems associated with 12½ Hz judder in areas of revealed background in the decoded picture may be reduced, as the sampling structure would no longer be divided into fixed groups of four fields. This may improve the worst-case performance of the system, for example on picture material such as the SNOOKER and CAPTION test sequences. The use of a 'rolling' sampling structure would also reduce the number of frame stores required in the decoder.

A less radical approach would be to examine the use of more sophisticated compatibility filtering in order to reduce the judder visible in the compatible picture. Some initial attempts at using temporal filtering in moving picture areas only succeeded in exchanging judder for the formation of multiple images; however the use of *motion-compensated* compatibility filtering should enable judder and dot patterning to be reduced in all areas sent via the 80 ms branch, without the introduction of multiple images. Such an approach would require slightly more sophisticated hardware in the decoder.

The use of a 40 ms motion-compensated branch in addition to a motion-compensated 80 ms branch may improve both the quality of the compatible and the decoded picture. Such a branch could be used to good effect in sequences such as CAPTION and SNOOKER as discussed in Section 5.2.1. However, there may be additional switching artifacts introduced by the use of a third branch. The decoder complexity may also be increased, although it may prove possible to adapt the 80 ms branch decoder hardware to deal with both branches simultaneously. The compatible picture could be improved by using a 40 ms branch instead of the 80 ms branch in regions of high velocity, where the additional resolution provided by the 80 ms branch may be of little use.

7. CONCLUSIONS

This Report has described research into the application of motion compensation to bandwidth reduction systems based on adaptive sub-sampling.

A number of conclusions can be drawn from this work:

(1) The use of motion compensation can significantly improve the performance of this kind of bandwidth system by enabling moving objects to be transmitted with high spatial resolution. A reduction in mean impairment grade from 1.34 to 0.59 was achieved in subjective tests.

(2) Motion compensation introduces additional impairments in the compatible picture that take the form of motion judder. The use of an 80 ms motion-compensated branch resulted in 12½ Hz judder that was quite objectionable.

(3) A small amount of 12½ Hz judder was also visible in the decoded pictures in areas of revealed background or where two detailed objects with different motion vectors fell into the same block in the picture. This was the only serious artefact produced.

(4) Comparison with systems that incorporated a 40 ms motion-compensated branch instead of an 80 ms one indicated that the use of the latter gave a worthwhile improvement in resolution in areas of tracked motion. However, systems incorporating a 40 ms motion-compensated branch tend to introduce artefacts in both the compatible and decoded pictures at 25 Hz rather than 12½ Hz, the higher frequency making them less objectionable.

The algorithm described here was not selected for implementation in hardware by the Eureka 95 High Definition Television project, despite giving the highest quality decoded pictures when compared to all other proposed systems. The main reason for this was the low quality of the compatible picture, together with the requirement for more frame stores in the decoder than for other systems.

Nevertheless, there are a number of other possible applications for this kind of bandwidth reduction system. Suggestions have been made for further research to determine whether a bandwidth reduction system can be devised that overcomes

the problems associated with an 80 ms motion-compensated branch without reducing the resolution in moving areas as is the case with a 40 ms branch.

UK patents have been applied for that cover the principles of the bandwidth reduction processing[13], the details of the algorithm described in this Report[14], and the motion estimation technique[15]. Equivalent applications have also been filed in other countries.

8. ACKNOWLEDGEMENTS

The author would like to thank S.C. Dabner for organising the subjective tests at Kingswood Warren, and R. Storey and T.J. Borer for helpful discussions. The help of colleagues at other organisations involved in the Eureka 95 project, who provided test sequences and organised the subjective tests in other laboratories, is also gratefully acknowledged.

9. REFERENCES

1. STOREY, R. 1986. HDTV Motion adaptive bandwidth reduction using DATV. BBC Research Department Report No. BBC RD 1986/5.

2. KNEE, M.J., WELLS, N.D. 1988. Bandwidth compression for HDTV broadcasting: Investigation of some adaptive subsampling strategies. BBC Research Department Report No. BBC RD 1988/9.

3. ROBERTS, A. 1983. Improved display of 625-line television pictures. BBC Research Department Report No. BBC RD 1983/8.

4. SEYLER, A.J., BUDRIKIS, Z.L. 1965. Detail perception after scene changes in television image presentations. IEEE Transactions on Information Theory, Vol. IT-11, 1965, pp. 31-43.

5. THOMAS, G.A. 1987. Television motion measurement for DATV and other applications. BBC Research Department Report No. BBC RD 1987/11.

6. PEARSON, J.J., HINES, D.C., GOLOSMAN, S., KUGLIN, C.D. 1977. Video-rate image correlation processor. SPIE Vol. 119. Application of Digital Image Processing (IOCC 1977), pp. 197-205.

7. GIROD, B., THOMA, R. 1985. Motion-compensating field interpolation from interlaced and non-interlaced grids. Conference on Image Coding, 4 - 6 Dec. 1985, Cannes, France. Proc. SPIE Int. Soc. Opt. Eng. (USA) Vol. 594, pp. 186-193, 1986.

8. WESTON, M. 1988. Fixed, adaptive and motion compensated interpolation of interlaced TV pictures. Second International Workshop on Signal Processing for HDTV, 29th Feb. - 2nd March 1988, L'aquila, Italy.

9. HURALT, J-P., ARRAGON, J-P. 1988. Development of advanced HD-MAC coding algorithms. IBC 88, IEE Conference Publication No. 293, Sept. 1988, pp. 54-56.

10. VREESWIJK, F.W.P. et. al. 1988. HD-MAC coding of high definition television signals. IBC 88, IEE Conference Publication No. 293, Sept. 1988, pp. 62-65.

11. ARRAGON, J-P., FONSALAS, F., HAGHIRI, M. 1988. Motion compensated interpolation techniques for HD-MAC. IBC 88, IEE Conference Publication No. 293, Sept. 1988, pp. 70-73.

12. PELE, D., CHOQUET, B. 1988. Estimation and segmentation of apparent motion fields in HD-MAC. IBC 88, IEE Conference Publication No. 293, Sept. 1988, pp. 74-77.

13. UK Patent Application GB2188509A, 'Video signal processing for bandwidth reduction', Sept. 1987.

14. UK Patent Specification No. 8814822.6, 'Bandwidth reduction system for television', June 1988.

15. UK Patent Application GB2188510A, 'TV picture motion measurement', Sept. 1987.

APPENDIX 1

Details of the Motion Vector Estimation Algorithm

The technique of motion vector estimation chosen for this work involved performing a phase correlation between fairly large blocks in two successive pictures, and extracting the dominant vectors present by locating peaks in the correlation surface. The images were then displaced relative to each other by each dominant vector in turn, and the match error computed. The error was spatially filtered and sampled at the centres of each small block in the picture. The vector that gave the smallest match error was then assigned.

The development of this technique is described in Ref. 5. It has been adapted slightly for use in this application; the details are given below. The principal modifications were that the video signal was sub-sampled prior to motion estimation, and that the calculation of assignment errors was performed over periods of both 20 ms and 40 ms. The first of these modifications was intended to reduce the amount of hardware required to implement the process on an HDTV signal. The purpose of the second modification was to improve the reliability of the vector assignment process.

The technique as described below was used in the bandwidth reduction system based on 12½ Hz transmission, described in Section 4 of this Report. However, the technique used in the moving sampling structure system described in Section 3 was very similar.

A1.1 Initial sub-sampling of picture

The incoming video signal was pre-filtered both horizontally and vertically (in a field) with the 7-tap FIR filter whose coefficients are given in Table 6. In the case of signals from 25 Hz film, the vertical filter was applied to picture lines rather than field lines (it was assumed that an indication of whether the signal originated from film or video camera was available). The filtered sequence was orthogonally sub-sampled to produce a quarter-size picture.

In the case of interlaced video, this process was rather primitive; it generated the second of the two sub-sampled fields one HDTV picture line higher than it should be, and left significant residual aliasing in areas of high vertical detail. A more sophisticated filter could have been used, although earlier work has shown that each field used in the phase correlation process should only contain contributions from one input field otherwise spurious peaks can be introduced.

A1.2 Measurement of dominant vectors

The odd fields of each quarter-size sub-sampled picture (which would contain 720 × 288 samples if the high definition source contained 1440 × 1152) were divided up into blocks of 32 pixels by 16 field lines, with an overlap of 4 pixels horizontally so that there was an integral number of blocks across the picture. The block spacing was thus 30 pixels horizontally and there was a total of 24 × 18 blocks in each picture.

Each block was multiplied by a windowing function which was unity over the whole block except for the first and last three pixels in a row/column, where it dropped to 2/3, 1/3 and 0 as the edge of the block was approached. This window was sufficient to reduce edge effects without unduly reducing the effective area of each block.

A phase correlation was performed between corresponding blocks in successive odd fields. This consisted of calculating the inverse Fourier transform of Z, where

$$Z(m,n) = \frac{\mathrm{G1}(m,n)\mathrm{G2}^{*}(m,n)}{|\,\mathrm{G1}(m,n)\mathrm{G2}^{*}(m,n)\,|}$$

where G1, G2 are the Fourier transforms of blocks in successive odd fields,

m,n are horizontal and vertical frequencies, and

* represents the complex conjugate.

The resulting correlation surface was subjected to a first-order recursive temporal filter, such that each point on the surface was set to 0.7 times its initial value plus 0.3 times the value of the point in the preceding filtered correlation surface. The object of the filtering was to reduce the amplitude of peaks due to noise; peaks due to real movement tend not to move significantly from one frame to the next.

The filtered correlation surface was searched to locate the highest three peaks in an area corresponding to velocities in the range ±15 pixels/picture period and ±4 picture lines per picture period in the reduced-size picture. In the full-size HDTV picture this represented a maximum velocity of 30 pixels and 8 picture lines per picture period (1 second per picture width; 5.75 seconds per picture height).

The peaks in the correlation surface were interpolated to estimate the sub-pixel part of the displacement. This interpolation was performed by fitting an inverted 'V' shape to the three nearest points independently in x and y (Fig. A1.1); this shape provided a good model of the peak profile produced by the correlation process.

No quantisation was applied to the resulting vector components (i.e. they were processed as real numbers), although they were unlikely to be accurate to more than 0.1 pixel in the reduced size picture. In the subsequent coding of motion vectors for transmission, quantisation to the nearest quarter of an HDTV pixel per picture period was assumed.

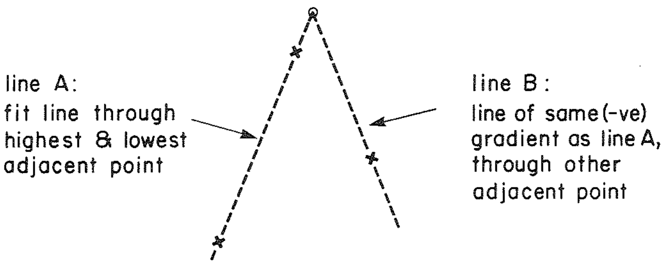

Fig. A1.1 - Inverted 'V' used for interpolation of the correlation surface.

× Points in correlation surface
o Interpolated peak location.

A1.3 Selection of trial vectors

A list of trial vectors was compiled for each block using vectors measured in the block itself and the surrounding blocks. Vectors were added to the list as long as they met the following criteria:

(a) The vector differs from all vectors already on the list by at least 0.1 pixel (or picture line) per picture period, on the scale of the reduced-size picture.

(b) The height of the peak for the vector in the correlation surface is at least 0.25 times the height of the largest peak in the current measurement block.

A maximum of 8 vectors were chosen, and they were selected with the following priority:

(1) The vectors measured in the current block (this was a maximum of three);

(2) the vectors with the highest peak in the adjacent blocks to the left, right, above and below;

(3) the vectors with the second highest peaks in these blocks;

(4) the vectors with the highest peaks in the diagonally adjacent blocks to the top left, bottom right, top right and bottom left;

(5) the vectors with the second highest peaks in these blocks.

In many cases, there were less that 8 vectors in the list after all the candidate vectors had been considered, using the selection rules (a) and (b) above. In this case, the list was left short.

A1.4 Motion vector assignment

Each small block was assigned one vector from the menu of the measurement block in which its centre lay. The blocks were 12 × 12 pixel diamonds in the original HDTV image (Fig. 6); 6 × 6 diamond-shaped blocks were used in the assignment process, since the image had been sub-sampled. The assignment process is illustrated in Fig. A1.2, and is described below.

The first stage of the assignment process was to perform an interlace-to-sequential conversion on every other odd down-filtered field, using a fixed vertical-temporal filter (Table A1.1). These conversions were hence performed at the rate of 12½ Hz. The resulting picture is referred to as the 'sequential field' in the following description. In the case of a video signal originating from 25 Hz film, the sequential field was formed using a pair of consecutive fields which together constitute a sequentially-scanned image. The sequential field corresponded to the transmitted image in the bandwidth reduction system described in Section 4.

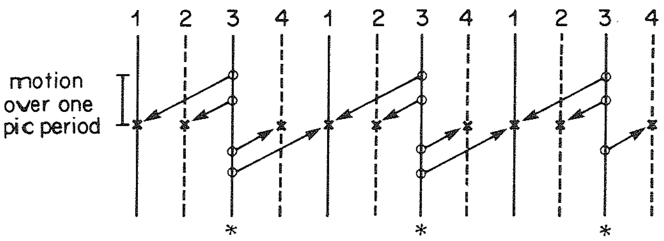

Fig. A1.2 - Overview of the vector assignment process.

* Sequential fields
× Pixels for which vectors are being assigned
o Interpolated pixels.

Table A1.1
Fixed interlace-to-sequential conversion filter used during vector assignment

+0.031 □		+0.031 □
	−0.026 □	
−0.116 □		−0.116 □
	+0.526 □	
+0.070 □	•	+0.170 □
	+0.526 □	
−0.116 □		−0.116 □
	−0.026 □	
+0.031 □		+0.031 □
preceding field	current field	following field

□ - pixels in original interlaced fields
• - pixel being interpolated

An assignment error was calculated at every pixel in odd field lines of the measurement block for each vector in the menu by subtracting the preceding odd field from a version of the sequential field displaced by the trial vector. A 4-point linear spatial interpolator was used to interpolate the sequential field to deal with sub-pixel vectors. The modulus of the resulting error signal was filtered with a simple rectangular-aperture filter of 9 pixels by 5 field lines in the quarter-size down-filtered sequence.

A second assignment error was calculated by repeating this process using the even field immediately preceding the sequential field. The trial vectors were scaled down by a factor of two and the different vertical position of the field lines was allowed for. In the case of signals from 25 Hz film, the factor of two scaling was not used because both fields referred to an instant of time one picture period before the sequential field.

The total assignment error for the vector was formed by adding together these two errors, and multiplying by a weighting factor. The weighting factor was always set to unity for the work described in this Report, although

subsequent work has shown that there is some benefit in having a factor that increases slightly with the modulus of the vector being tested. A suitable form of weighting factor was found to be

$$W = 1 + 0.05 * \mathrm{mod}(v)$$

where mod(v) is the magnitude of the motion vector in pixels per picture period in the sub-sampled picture. The object of the weighting factor is to bias assignment decisions in favour of small vectors when both large and small vectors produce similar assignment errors. This significantly reduces the occurrence of large random vectors in plain picture areas, and helps to resolve ambiguities caused by periodic structures.

The trial vector that gave the minimum total assignment error was selected. The magnitude of all vectors was multiplied by two in order to compensate for the initial sub-sampling operation, thereby generating vectors applicable to the original HDTV signal.

This process generated vectors which indicated the displacement that should be applied to the sequential field appropriate for the generation of the two preceding fields. The process was repeated for the two fields following the sequential field. Thus two sets of vectors were generated for use with each sequential field, one set referring to the preceding 1/25th of a second, the other set to the following 1/25th of a second. In the subsequent bandwidth reduction processing, linear motion was assumed within each period of 1/25th of a second.

The vector information generated as described here was thus appropriate for use in a 12½ Hz to 50 Hz up-conversion process, where each transmitted frame would be used to generate the two preceding fields and the two following fields.

APPENDIX 2

Description of the Test Sequences

The eight sequences used in the subjective tests each consisted of 100 *YUV* frames, of 720 pixels by 576 lines. All the sequences except KITCHEN were 2:1 interlaced. The sequences are briefly described below.

KEW (Kew Gardens)

A camera pan showing the front of *The Temperate House* in Kew Gardens. The sequence originated from 50 Hz film shot originally for use with an HDTV telecine, and hence contained a high amount of detail and interline twitter as it was scanned on a 625-line telecine. The temporal resolution was also very high, due to the short shutter time (about 1/100th of a second) of the film camera.

KITCHEN (Kitchen scene; also known as Kitchen Grass)

A sequence originated on 24 Hz film, showing a slow zoom into a scene showing a woman holding a plate of grass sitting in a kitchen. The background was slightly out of focus.

CAR (Car scene; also known as PRL Car)

A fast camera pan (about 1 second per picture width) following a car which decelerated to stationary during the sequence. The background was the outside of a building, the major features being window frames and Venetian blinds. The sequence was originated using an HDTV camera and down-filtered to give a 625-line picture.

SNOOKER (Snooker balls)

Two coloured snooker balls moving slowly against a highly detailed stationary background. (Electronic tube camera).

CAPTION (Rolling caption)

Electronically-generated blue rolling captions moving over a stationary detailed background showing a postage stamp.

RE1 (Renata 1)

A sequence from an electronic camera, showing a young woman walking in front of a detailed background (mainly showing a large calendar); the camera panned and zoomed slightly to follow the woman.

RE2 (Renata 2; also known as Renata Butterfly and Edit)

A slow zoom into a close-up of the face of the woman in the previous sequence, followed by a slow cross fade into a slow zoom onto a highly detailed picture of butterflies. (Electronic tube camera).

CACTUS (Cactus & Comb)

A slow pan and zoom over a cactus and a comb against a plain bright blue background. (Electronic tube camera).

Printed by BBC RESEARCH DEPARTMENT, Kingswood Warren, Tadworth, Surrey, KT20 6NP